

# Modeling an Airbnb Use-Case Database for Room

## Booking

Rahul Ragiri

Matriculation number: 32112978

Conception Phase

DLBDSPBDM01 - Project: Build a Data Mart in SQL

Tutor: Prof. Musharaf Doger

February 29, 2024

## 1. Project goals

The goal of this project is to provide a dependable, efficient, and user-friendly database for an app that operates similarly to Airbnb. This database will be used to handle and store data on properties, bookings, users, hosts, reviews, and other related entities. The database will be designed to support a variety of user roles and activities and will provide the information and capabilities these users require.

## 2. Requirement specification

### 2.1 Which roles exist among individuals or user groups?

**Guests:** These are the primary users of the platform who are trying to find accommodations. They can be locals going on a staycation, business travelers, or tourists. Their preferences range considerably, from low-cost to high-end accommodations, and from travelling alone to taking family trips.

**Hosts:** Hosts are people or organizations who list their houses for rent on the marketplace. These might be property managers overseeing several rental homes, or homeowners who rent out portions or their whole house. They provide a variety of lodging options, including luxurious and affordable lodging as well as unusual stays in yurts or treehouses.

**Airbnb Admins:** These are the people in charge of running the platform. They make sure that the platform functions properly, that all transactions are safe, and that hosts and guests abide by the company's policies. Their duties might range widely, including dispute settlement, platform security, and customer service.

## 2.2 What operations are carried out by these roles?

**Guest:** guests search for and book accommodation that meets their needs, and they have the option to filter properties according to several criteria. In addition, they may give evaluations and ratings after their visit and contact the hosts with any questions or concerns.

**Hosts:** The hosts reply to guest inquiries for reservations and list their homes with all the information required. After the stay, they might choose to rate guests and write a review. They are also accountable for maintaining the accuracy of their property listings.

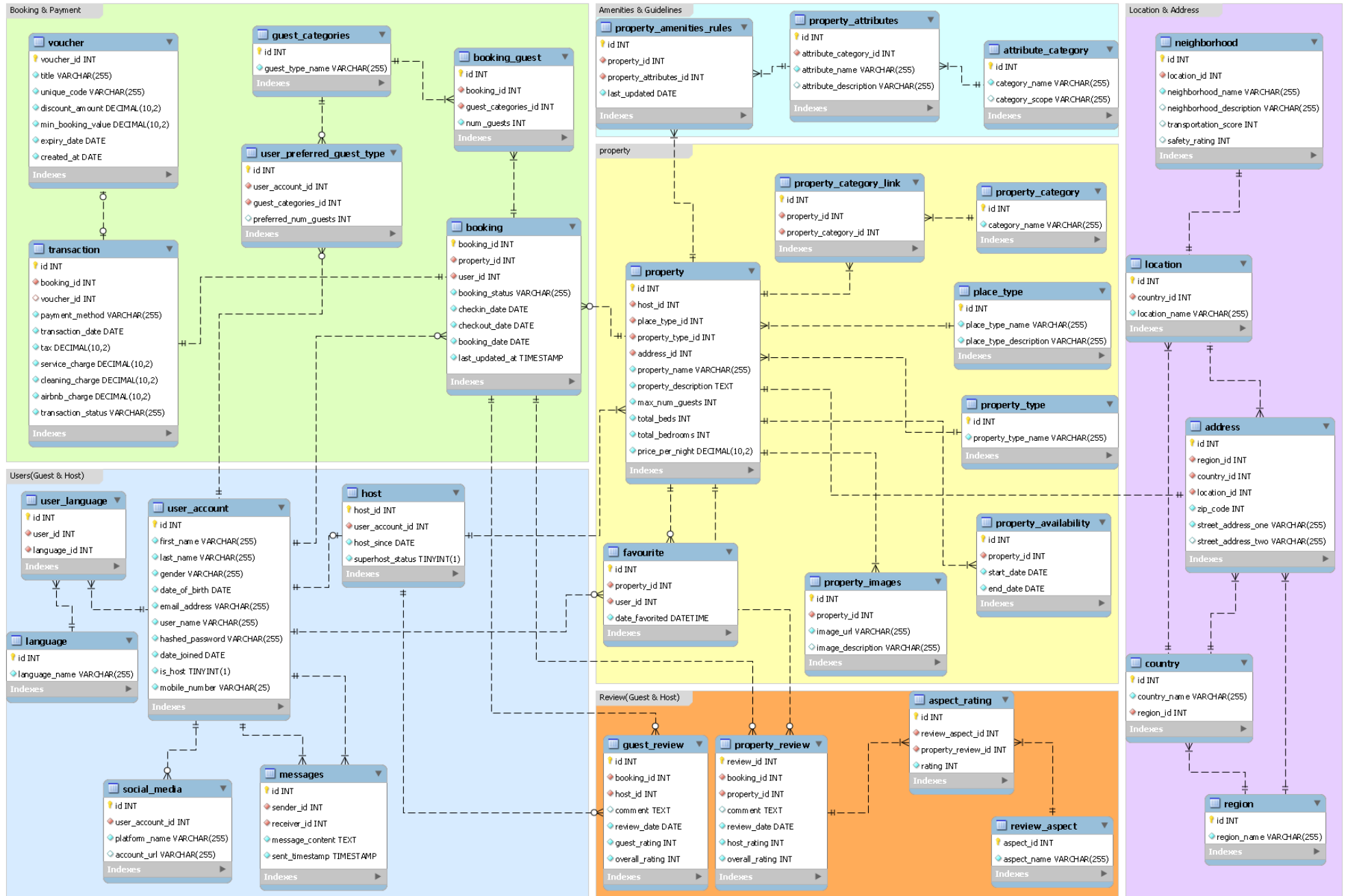
**Airbnb admin:** Admins manage platform activities, handle user accounts, settle conflicts, and impose regulations. To make wise decision-making for improving the platform, they examine platform data.

## 2.3 Which data and functionalities are necessary?

**Data required:** Information about users (both guests and hosts), properties, reservations, reviews, communications, and transactions must be stored on the platform. User favorite, account information, and personal data are examples of user data. Property information consists of specifics of the property, its features, availability, price, and property review. Information on the guest, the host, the property, and the booking are all included in the booking data. Review information consists of comments and ratings left by both hosts and guests. Guest and host communications are included in the message data. Information regarding the payment, the reservation, and the transaction's status are all included in the transaction data.

**Functions required:** Users must be able to manage their accounts, search, and filter properties, make and manage reservations, post reviews, interact with other users, and make payments via the platform. hosts should have the ability to list their properties, set availability, and communicate with potential guests. Guests should also be able to view host profiles, property images, and property reviews to make informed booking decisions. Additionally, Admins must also be able to manage user accounts, resolve conflicts, enforce platform guidelines, and do data analysis on the platform.

### 3. EER(Enhanced-Entity-Relationship)



# Schema Report for Database: airbnb

## Table List

- [aspect\\_rating](#)
- [booking](#)
- [booking\\_guest](#)
- [country](#)
- [attribute\\_category](#)
- [favourite](#)
- [guest\\_categories](#)
- [host](#)
- [guest\\_review](#)
- [language](#)
- [location](#)
- [messages](#)
- [place\\_type](#)
- [property](#)
- [property\\_availability](#)
- [property\\_category](#)
- [property\\_category\\_link](#)
- [property\\_attributes](#)
- [property\\_amenities\\_rules](#)
- [property\\_images](#)
- [property\\_review](#)
- [property\\_type](#)
- [region](#)
- [review\\_aspect](#)
- [social\\_media](#)
- [transaction](#)
- [user\\_account](#)
- [user\\_language](#)
- [voucher](#)
- [address](#)
- [neighborhood](#)
- [user\\_preferred\\_guest\\_type](#)

Table: aspect\_rating

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each record
review_aspect_id	INT	Yes	No	Yes		Identifier of the review aspect
property_review_id	INT	Yes	No	Yes		Identifier of the property review
rating	INT	Yes	No	No		Rating given to the aspect in the review

Table: booking

Name	Data Type	Not Null	PK	FK	Default	Comment
booking_id	INT	Yes	Yes	No		Unique identifier of booking
property_id	INT	Yes	No	Yes		Identifier of booked property
user_id	INT	Yes	No	Yes		Identifier of user who made the booking
booking_status	VARCHAR(255)	Yes	No	No		Status of booking
checkin_date	DATE	Yes	No	No		Check-in date of guest
checkout_date	DATE	Yes	No	No		Check-out date of guest
booking_date	DATE	Yes	No	No		Date when booking was made
last_updated_at	TIMESTAMP	Yes	No	No	CURRENT_TIMESTAMP	tracks the last time a record was updated

Table: booking\_guest

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier of booking guest
booking_id	INT	Yes	No	Yes		identifier of associated booking

guest_categories_id	INT	Yes	No	Yes		Identifier of guest demographics (ex: adults, childrens, infants, pets, etc.)
num_guests	INT	Yes	No	No		Number of guests of a specific guest type (ex: adult 2, children 2, infant 1, etc.)

Table: country

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier for each country
country_name	VARCHAR(255)	Yes	No	No		Name of the country, must be unique
region_id	INT	Yes	No	Yes		Identifier of the region of the country

Table: attribute\_category

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each detail category
category_name	VARCHAR(255)	Yes	No	No		Name of the detail category
category_scope	VARCHAR(255)	No	No	No	NULL	Brief description of the attribute category's name (ex: "Amenities - Facilities within the property")

Table: favourite

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier for each review
property_id	INT	Yes	No	Yes		Identifier of the property
user_id	INT	Yes	No	Yes		Identifier of the user
date_favorited	DATETIME	Yes	No	No		Date and time the property was marked as favourite

Table: guest\_categories

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier for each guest type
guest_type_name	VARCHAR(255)	Yes	No	No		Name of the guest type(ex: adult, children, infant, etc.)

Table: host

Name	Data Type	Not Null	PK	FK	Default	Comment
host_id	INT	Yes	Yes	No		Unique identifier of host
user_account_id	INT	Yes	No	Yes		Identifier of user account associated with host
host_since	DATE	Yes	No	No		Date when user became a host
superhost_status	TINYINT(1)	Yes	No	No		Boolean indicating superhost status

Table: guest\_review

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of the review
booking_id	INT	Yes	No	Yes		Identifier of the booking associated with review
host_id	INT	Yes	No	Yes		Foreign key referencing the host table
comment	TEXT	No	No	No	NULL	Text of the review written by the host
review_date	DATE	Yes	No	No		Date when the review was posted
guest_rating	INT	Yes	No	No		Numerical rating given to guest
overall_rating	INT	Yes	No	No		Rating given by the host to the guest

Table: language

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier for each language
language_name	VARCHAR(255)	Yes	No	No		Name of the language, must be unique

Table: location

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier for each location
country_id	INT	Yes	No	Yes		Identifier of the country of the location
location_name	VARCHAR(255)	Yes	No	No		Name of the location, must be unique

Table: messages

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		unique identifier for each message
sender_id	INT	Yes	No	Yes		identifier of the user who sent the message
receiver_id	INT	Yes	No	Yes		Identifier of the user who received the message
message_content	TEXT	Yes	No	No		Content of the message
sent_timestamp	TIMESTAMP	Yes	No	No		Time when the message was sent

Table: place\_type

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier of place type
place_type_name	VARCHAR(255)	Yes	No	No		Name of place type
place_type_description	VARCHAR(255)	Yes	No	No		Description of place type



Table: property

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each record
host_id	INT	Yes	No	Yes		Foreign key referencing the host table
place_type_id	INT	Yes	No	Yes		Foreign key referencing the place_type table
property_type_id	INT	Yes	No	Yes		Foreign key referencing the property_type table
address_id	INT	Yes	No	Yes		Foreign Key referencing address table
property_name	VARCHAR(255)	Yes	No	No		Name of the property
property_description	TEXT	Yes	No	No		Description of the property
max_num_guests	INT	Yes	No	No		Maximum number of guests that can be accommodated in a property
total_beds	INT	Yes	No	No		Total number of beds, must be greater than 0
total_bedrooms	INT	Yes	No	No		Total number of bathrooms, must be greater than 0
price_per_night	DECIMAL(10,2)	Yes	No	No		Price per night, must be greater than 0

Table: property\_availability

Name	Data Type	Not Null	PK	FK	Default	Comment
Id	INT	Yes	Yes	No		Unique identifier for each availability entry
property_id	INT	Yes	No	Yes		Identifier of the property
start_date	DATE	Yes	No	No		Start date of the availability period
end_date	DATE	Yes	No	No		End date of the availability period

Table: property\_category

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each property category
category_name	VARCHAR(255)	Yes	No	No		Name of the property category, must be unique

Table: property\_category\_link

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each record
property_id	INT	Yes	No	Yes		Identifier of the property
property_category_id	INT	Yes	No	Yes		Identifier of the property category

Table: property\_attributes

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of property details
attribute_category_id	INT	Yes	No	Yes		Identifier of attribute category table
attribute_name	VARCHAR(255)	Yes	No	No		Name of attribute (ex: amenities-swimming pool, house rules-no smoking, safety & property-fire extinguisher)
attribute_description	VARCHAR(255)	No	No	No	NULL	Description of the attribute

Table: property\_amenities\_rules

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each property feature link
property_id	INT	Yes	No	Yes		Identifier of the property

property_attributes_id	INT	Yes	No	Yes		Identifier of the property attributes
last_updated	DATE	Yes	No	No		Date the amenities/guidelines record was last updated

Table: property\_images

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of the image
property_id	INT	Yes	No	Yes		Foreign key referencing unique id of the property table
image_url	VARCHAR(255)	Yes	No	No		URL of the image
image_description	VARCHAR(255)	No	No	No	NULL	Description of the image

Table: property\_review

Name	Data Type	Not Null	PK	FK	Default	Comment
review_id	INT	Yes	Yes	No		Unique identifier for each review
booking_id	INT	Yes	No	Yes		Foreign key referencing the booking table
property_id	INT	Yes	No	Yes		identifier of the property being reviewed by booking(user)
comment	TEXT	No	No	No	NULL	Comment written by the user in the review
review_date	DATE	Yes	No	No		Date when the review was written
host_rating	INT	Yes	No	No		Numerical rating given to host
overall_rating	INT	Yes	No	No		Overall rating given by the user in the review, between 1 to 5

Table: property\_type

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each property type
property_type_name	VARCHAR(255)	Yes	No	No		Name of the property type

Table: region

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each region
region_name	VARCHAR(255)	Yes	No	No		Name of the region, must be unique

Table: review\_aspect

Name	Data Type	Not Null	PK	FK	Default	Comment
aspect_id	INT	Yes	Yes	No		Unique identifier for each review aspect
aspect_name	VARCHAR(255)	Yes	No	No		Name of the review aspect, must be unique

Table: social\_media

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each social media
user_account_id	INT	Yes	No	Yes		Identifier of the user who owns the social media
platform_name	VARCHAR(255)	Yes	No	No		Name of the social media platform
account_url	VARCHAR(255)	No	No	No	NULL	URL of the social media account, must be unique

Table: transaction

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier for each transaction
booking_id	INT	Yes	No	Yes		Identifier of the booking related to the transaction

voucher_id	INT	No	No	Yes	NULL	Identifier of the voucher used in the transaction
payment_method	VARCHAR(255)	Yes	No	No		Payment method used in the transaction
transaction_date	DATE	Yes	No	No		Date when the transaction was made
tax	DECIMAL(10,2)	Yes	No	No		Tax amount in the transaction
service_charge	DECIMAL(10,2)	Yes	No	No		Service charge in the transaction
cleaning_charge	DECIMAL(10,2)	Yes	No	No		Cleaning charge in the transaction
airbnb_charge	DECIMAL(10,2)	Yes	No	No		Airbnb charge in the transaction
transaction_status	VARCHAR(255)	Yes	No	No		Status of the transaction

Table: user\_account

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of user account
first_name	VARCHAR(255)	Yes	No	No		First name of the user
last_name	VARCHAR(255)	Yes	No	No		Last name of the user
gender	VARCHAR(255)	Yes	No	No		Gender of the user
date_of_birth	DATE	Yes	No	No		Date of birth of the user
email_address	VARCHAR(255)	Yes	No	No		Email address of the user
user_name	VARCHAR(255)	Yes	No	No		Username chosen by the user
hashed_password	VARCHAR(255)	Yes	No	No		Hashed and salted representation of the users password
date_joined	DATE	Yes	No	No		Date when the user joined
is_host	TINYINT(1)	Yes	No	No		Boolean indicating if the user is a host
mobile_number	VARCHAR(25)	Yes	No	No		Contact number for communication and verification

Table: user\_language

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of user-language link
user_id	INT	Yes	No	Yes		Identifier of associated user
language_id	INT	Yes	No	Yes		Identifier of associated language

Table: voucher

Name	Data Type	Not Null	PK	FK	Default	Comment
voucher_id	INT	Yes	Yes	No		Unique identifier of voucher
title	VARCHAR(255)	Yes	No	No		Title of voucher
unique_code	VARCHAR(255)	Yes	No	No		Unique code of voucher
discount_amount	DECIMAL(10,2)	Yes	No	No		Discount amount provided by voucher
min_booking_value	DECIMAL(10,2)	Yes	No	No		Minimum booking value for voucher applicability
expiry_date	DATE	Yes	No	No		Expiry date of voucher
created_at	DATE	Yes	No	No		Creation Date of voucher

Table: address

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		unique identifier for each address record.
region_id	INT	Yes	No	Yes		Region or administrative division
country_id	INT	Yes	No	Yes		Country associated with the address
location_id	INT	Yes	No	Yes		Specific location (e.g., city)
zip_code	INT	Yes	No	No		Postal code or ZIP code.
street_address_one	VARCHAR(255)	Yes	No	No		Primary part of the address (street name and house/building number).
street_address_two	VARCHAR(255)	No	No	No	NULL	secondary part of the address, can be NULL

Table: neighborhood

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of neighborhood
location_id	INT	Yes	No	Yes		Foreign key referencing the location table
neighborhood_name	VARCHAR(255)	Yes	No	No		Name of the neighborhood
neighborhood_description	VARCHAR(255)	No	No	No	NULL	Optional description of the neighborhood
transportation_score	INT	No	No	No	NULL	Optional score for the availability and convenience of public transport, rated out of 10 (integer)
safety_rating	INT	No	No	No	NULL	Optional rating for the perceived safety of the neighborhood, rated out of 10 (integer)

Table: user\_preferred\_guest\_type

Name	Data Type	Not Null	PK	FK	Default	Comment
id	INT	Yes	Yes	No		Unique identifier of user preferred guests type
user_account_id	INT	Yes	No	Yes		Foreign key referencing the user in user account table
guest_categories_id	INT	Yes	No	Yes		Foreign key referencing the guest categories table (ex: adult, children, infant)
preferred_num_guests	INT	No	No	No	NULL	Number of guests of the preferred type

### **3. Challenges and Solutions in Database Design**

In this project, we undertook the task of creating a database for a hotel booking business inspired by Airbnb. The primary challenge was distinguishing between guests and hosts, as users could fulfill both roles. To address this, we established a separate table for hosts to manage their roles independently from guest accounts.

Host reviews played a crucial role, setting Airbnb apart from other rental booking services. To ensure accuracy, we linked reviews directly to individual guest bookings through the booking table. Initially, there was a dilemma regarding whether to connect the `property_reviews` and `guest_reviews` tables to either the booking or the property table. However, connecting them to the property table would have led to redundancy and inconsistency. This decision ensured that hosts could only review guests who had stayed at their property.

To maintain data integrity, we implemented trigger functions. These functions prevented hosts from self-booking or self-reviewing their own properties, thus ensuring fair and accurate reviews.

Regarding the calculation of the total price, there was uncertainty about whether a `total_price` attribute was necessary in the booking table. Upon further examination of the schema, it became evident that we could dynamically compute the total price using additional relevant attributes. This approach not only guaranteed consistency but also reduced redundancy within the database structure.



# PROJECT: BUILD A DATAMART IN SQL (DLBDSPBDM01)

**A Use-Case of booking a hotel room: Airbnb**

---

**Presented by R. Rahul**  
**Development Phase**

# Overview

01- Introduction

02- SQL Operations: Table Creation and use-case, Data Insertion

03- Test-case

## **01- Introduction**

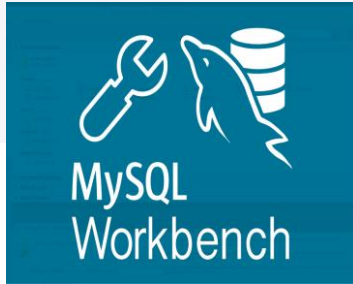
**Abstract, WorkSpace & Tools**

## Abstract

In this project, I applied my SQL skills to construct a database that mirrors the functionality of an Airbnb-like platform. The process began with the creation of an Entity-Relationship Diagram, which served as a visual blueprint for our database schema. The diagram provided a clear picture of the database schema and the relationships between different entities. Tables were designed to encapsulate a wide array of data, including details about users, properties, transactions, bookings, and reviews. To ensure the robustness and efficiency of the database, a series of test cases were developed and executed. The outcomes of these tests were meticulously documented to provide a comprehensive view of the database's structure and its capabilities. Along with the test results, illustrative screenshots of the database system were taken to show how everything works. These screenshots serve as a visual guide to the inner workings of the database. After testing and documentation, the database is now ready to be used in an Airbnb-like application.



## Workspace & Tools



### MySQL Workbench

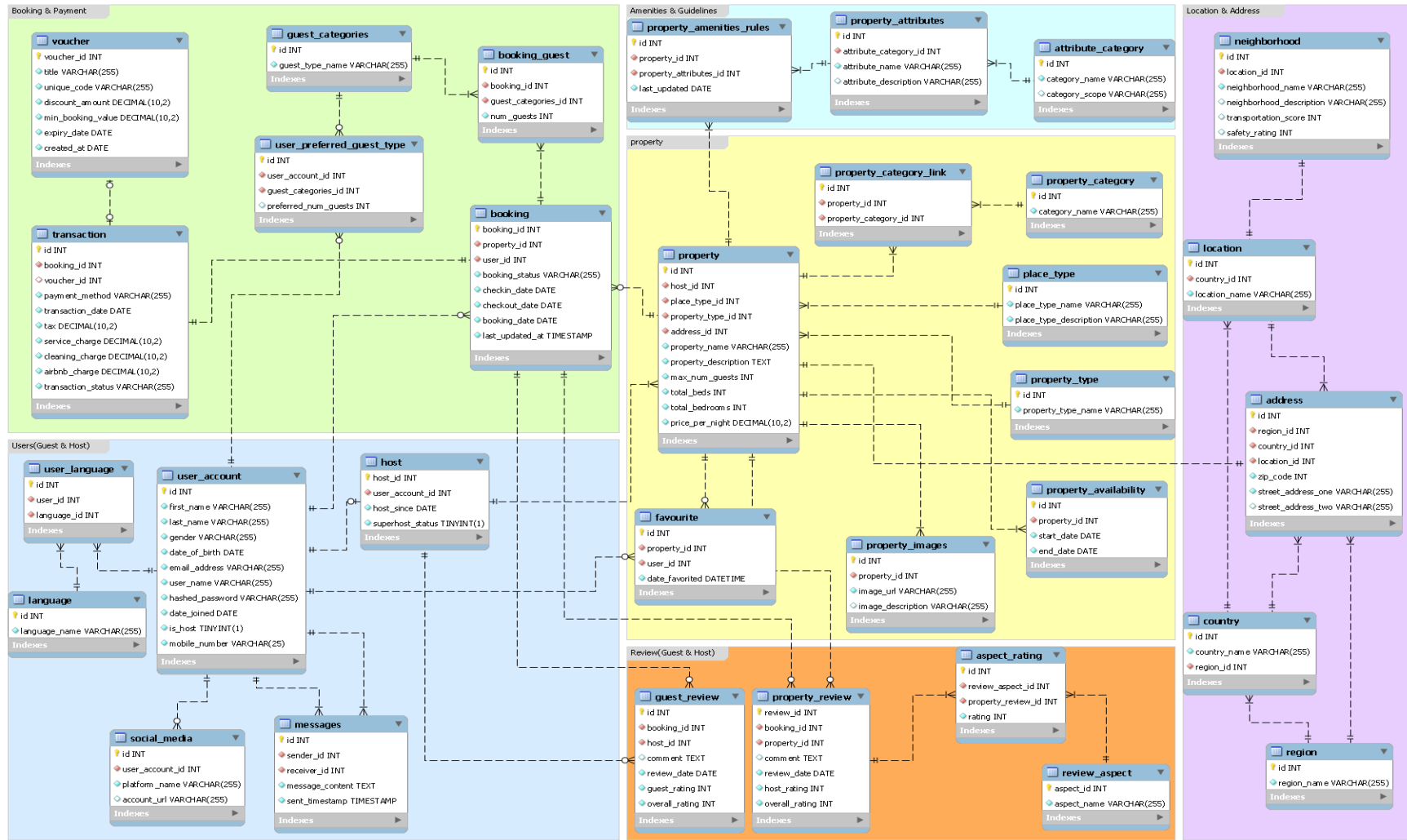
For this project, I selected tools that would streamline the process of designing and managing the database. I used MySQL Workbench for database management, which is known for its user-friendly interface and extensive community support.



### dbForge Studio for MySQL

To populate the database with dummy data, I made use of dbForge Studio because of its extensive features for data generation. In addition to this, I also manually entered some data into the database.

# EER (Enhanced-Entity-Relationship) diagram





## **02- SQL Operations: Table Creation and use-case, Data Insertion**

**Schema Implementation, Data  
Population, Functionality  
Verification**

## Schema Creation

We use a system called the "airbnb" schema to design our database. This schema essentially acts as a map for all the information stored in the database.

```
-- -----  
-- Schema airbnb  
-- -----
```

```
-- -----  
-- Schema airbnb  
-- -----
```

```
CREATE SCHEMA IF NOT EXISTS `airbnb` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci ;  
USE `airbnb` ;
```



# review\_aspect

```
-- Table `airbnb`.`review_aspect`  
  
DROP TABLE IF EXISTS `airbnb`.`review_aspect` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`review_aspect` (  
  `aspect_id` INT NOT NULL AUTO_INCREMENT,  
  `aspect_name` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`aspect_id`),  
  UNIQUE INDEX `aspect_name_UNIQUE` (`aspect_id` ASC) VISIBLE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
4 • SELECT  
5 *  
6 FROM  
7 review_aspect;
```

aspect_id	aspect_name
1	Outdoor Space
2	Kitchen
3	Location
4	Safety
5	Cleanliness
6	Noise Level
7	Comfort
8	Overall Experience
9	Check-in
10	Privacy
11	Value
12	View
13	Internet
14	Accuracy
15	Amenities
16	Communication
17	Accessibility
18	Bathroom
19	Bed
20	Guest Interaction

This table stores the different aspects that guests can rate in their reviews of properties (e.g., cleanliness, location, value).

**This query selects all aspects listed for property reviews.**

# user\_account

```
-- Table `airbnb`.`user_account`

DROP TABLE IF EXISTS `airbnb`.`user_account` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`user_account` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `first_name` VARCHAR(255) NOT NULL,
  `last_name` VARCHAR(255) NOT NULL,
  `gender` VARCHAR(255) NOT NULL,
  `date_of_birth` DATE NOT NULL,
  `email_address` VARCHAR(255) NOT NULL,
  `user_name` VARCHAR(255) NOT NULL,
  `hashed_password` VARCHAR(255) NOT NULL,
  `date_joined` DATE NOT NULL,
  `is_host` TINYINT(1) NOT NULL,
  `mobile_number` VARCHAR(25) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `user_name_UNIQUE` (`user_name` ASC) VISIBLE,
  UNIQUE INDEX `email_address_UNIQUE` (`email_address` ASC) VISIBLE,
  UNIQUE INDEX `mobile_number_UNIQUE` (`mobile_number` ASC) VISIBLE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

11																				
12	SELECT * FROM user_account;																			
13																				
null Grid																				
Edit Filter Rows Export(Report) Wrap Cell Contents																				
id	first_name	last_name	gender	date_of_birth	email_address	user_name	hashed_password	date_joined	is_host	mobile_number										
1	Lore	Keeton	Female	1974-07-28	LoreKeeton6138@gmail.com	Cleveland632	9f58c9201262c2818x44W39K...	2007-12-25	0	+33 666145421										
2	Hector	Alvarez	Male	1990-06-22	HectorAlvarez5371@gmail.com	Gene1995	C719417PU14Y21W17Q1N849D5...	2014-06-01	1	+33 4879738012										
3	Dawne	Eng	Female	1958-05-01	DawneEng5613@gmail.com	Adena615	LD25F4V3S38C06670444VEV2U...	2008-09-10	0	+81 1152165987										
4	Helen	Matson	Male	1957-07-16	HelenMatson7394@gmail.com	Keefe447	94UR43KQXQ107F9B992DE	2013-06-16	1	+91 9471295304										
5	Porfirio	Haggard	Male	1996-08-08	PorfirioHaggard3148@gmail.com	Marsela1992	07W9GKM18UW7Q19L564NDUUA...	2008-12-07	1	+44 3746499518										
6	Basil	Keith	Male	1984-05-17	BasilKeith7394@gmail.com	Upton1968	D0XK3694L75T5R8FID30T5N895...	2007-02-03	1	+44 8675583290										
7	Jana	Engel	Male	1961-12-08	JanaEngel1590@gmail.com	Abbe999	NXK1434B296F383U5V099770M0...	2009-08-29	0	+17 225139465										
8	Ethel	Haggerty	Female	1994-12-11	EthelHaggerty6844@gmail.com	Singletary85	1SASPIC24RE97LU068082334PK...	2014-10-24	1	+14 209840731										
9	Made	Matteson	Female	1954-02-02	MadeMatteson0342@gmail.com	Trinh3	BMV73F3VNTD5386D5324693F1...	2018-03-08	0	+81 5506951023										
10	Davis	Thibodeaux	Male	1978-01-25	DavisThibodeaux0363@gmail.com	Alayna76	N605703SASWLBW868W57797...	2018-05-30	1	+1 981632159										
11	Abby	Kalcher	Female	1961-11-15	AbbyKalcher8983@gmail.com	Ron727	X571GIC	2008-04-26	0	+33 431843678										
12	Tristan	Matthew	Male	1954-01-04	TristanMatthew7534@gmail.com	Reich26	2903G8583R259CSYN522L7U67...	2011-06-17	0	+49 1803668432										
13	Yolanda	England	Female	1956-07-24	YolandaEngland5595@gmail.com	Alta6	IAYZ76VEOP30227E23BWCO8P2...	2020-03-22	1	+49 3662540189										
14	Bernita	Broadway	Male	1962-03-23	BernitaBroadway4543@gmail.com	Emmie261	SC172380A90L3QV14VY115544...	2016-01-18	1	+44 6298080357										
15	Trent	Alves	Male	1986-03-03	TrentAlves2528@gmail.com	Alane11	DLQ06FB8884495G16Q368R80...	2016-01-21	0	+49 0351282961										
16	Johanna	Hahn	Male	1996-02-24	JohannaHahn7553@gmail.com	Gregory3	J	2010-10-23	1	+91 7311881642										
17	Leslie	Keller	Male	1954-03-15	LeslieKeller1270@gmail.com	Myron2021	0UX1SP5VHKC96JF2VDVDEBU...	2016-10-22	0	+16 755095814										
18	Damon	Sander	Male	1980-07-30	DamonSander5002@gmail.com	Deangelis325	X5STQ244H48H4NQ5VUE07YOIS...	2017-05-17	0	+33 6012336590										
19	German	North	Male	1969-05-24	GermanNorth1953@gmail.com	Lapointe91	FZV9569/QJSC2Y75U7U	2010-01-22	0	+44 363429172										
20	Adelaide	Thiel	Male	1954-01-19	AdelaideThiel0060@gmail.com	Aquino27	7G79039HQ8Q8WDM1BG25XL2...	2012-07-18	0	+44 7576646385										
21	Ellis	Sanders	Male	1972-09-13	EllisSanders1843@gmail.com	Janssen1980	5KY16VUK51Y4VR1BHMJ23V78F...	2007-01-04	1	+15 206395041										
22	Love	Matthews	Female	1954-01-30	LoveMatthews5324@gmail.com	Talisha1961	77	2021-03-30	0	+81 6072542873										
23	Romela	Kelley	Male	1954-03-05	RomelaKelley7033@gmail.com	Cher2017	U972969C5932W3K906B57R3R4L...	2007-12-24	0	+44 5990520618										
24	Alana	Roach	Male	1954-01-04	AlanaRoach1230@gmail.com	Willard2000	B2G38A057ULDFSH	2010-05-03	1	+19 920276354										
25	Lenny	Engle	Male	1957-07-18	LennyEngle4758@gmail.com	Mabele1975	HD2287W54BG5	2010-09-29	0	+49 4520492126										
26	Filberto	Mattingly	Male	1982-11-03	FilbertoMattingly2702@gmail.com	Dane836	36XK840959SLB4QOLEZ2158911	2007-01-09	1	+33 6885093241										
27	Domenic	Norton	Male	1970-03-04	DomenicNorton3598@gmail.com	Dara111	B1FP6K120MBE3V78312K074R2JL...	2007-12-11	1	+11 813725908										
28	William	Roark	Male	1975-06-17	WilliamRoark0458@gmail.com	Enoch2029	7H40UDSN46E342045W7V0J...	2009-11-08	1	+13 597035623										
29	Miles	Brook	Male	1963-02-06	MilesBrook5728@gmail.com	Brynur8	EZF7AK7SAQTYC16824274E3K...	2009-05-01	1	+49 5189992137										

This query retrieves all user account information from the database.

This table stores user account information for both guests and Airbnb hosts on the platform. It serves as the central table for user identities and links to various other tables based on user roles (guest or host).

# host

```
-- Table `airbnb`.`host`
-----

DROP TABLE IF EXISTS `airbnb`.`host` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`host` (
  `host_id` INT NOT NULL AUTO_INCREMENT,
  `user_account_id` INT NOT NULL,
  `host_since` DATE NOT NULL,
  `superhost_status` TINYINT(1) NOT NULL,
  PRIMARY KEY (`host_id`),
  INDEX `user_account_id` (`user_account_id` ASC) VISIBLE,
  UNIQUE INDEX `user_account_id_UNIQUE` (`user_account_id` ASC) VISIBLE,
  CONSTRAINT `host_ibfk_1`
    FOREIGN KEY (`user_account_id`)
      REFERENCES `airbnb`.`user_account` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
14 -- host
15 * SELECT
16   h.host_id,
17   CONCAT(u.first_name, ' ', u.last_name) AS host_name,
18   h.host_since,
19   h.superhost_status
20 FROM
21   host h
22 JOIN
23   user_account u ON h.user_account_id = u.id;
```

host_id	host_name	host_since	superhost_status
1	Hector Alvarez	2023-10-12	1
2	Helen Matson	2023-07-25	0
3	Porfirio Haggard	2023-05-12	0
4	Basel Keith	2023-07-17	1
5	Ethel Haggerty	2022-01-23	1
6	Davis Thibodeaux	2023-07-09	0
7	Yolando England	2023-02-22	0
8	Bernita Broadway	2023-01-12	0
9	Johanna Hahn	2022-06-15	0
10	Ellis Sanders	2022-04-20	1
11	Alana Roach	2022-02-15	0
12	Filberto Mattingly	2023-02-27	0
13	Domenic Norton	2022-06-05	0
14	William Roark	2022-11-13	1
15	Miles Brock	2022-01-23	1
16	Ashlea Amador	2022-07-26	1
17	Cristopher English	2023-06-07	1
18	Brandee Sanderson	2022-03-30	1
19	himadeep souja...	2023-09-16	1
20	krishnaveni Jinakala	2023-06-06	1
21	varenja emani	2023-07-07	0

Stores information about Airbnb hosts, linked to their user accounts.

This query retrieves host details (ID, full name, host since date, superhost status)

## place\_type

```
-- Table `airbnb`.`place_type`
-----

DROP TABLE IF EXISTS `airbnb`.`place_type` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`place_type` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `place_type_name` VARCHAR(255) NOT NULL,
  `place_type_description` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `place_type_name_UNIQUE` (`id` ASC) VISIBLE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
29 -- place_type
30 • SELECT
31     pt.place_type_name,
32     pt.place_type_description,
33     COUNT(*) AS num_guests
34 FROM
35     place_type pt
36 JOIN
37     property p ON pt.id = p.place_type_id
38 GROUP BY pt.place_type_name , pt.place_type_description;
39
```

place_type_name	place_type_description	num_guests
Studio	A compact apartment with an op...	1
Entire Apartment	A self-contained living space with a...	1
Entire House	A standalone house with multiple r...	2
Houseboat Cabin	A cozy cabin on a floating houseb...	1
Duplex	A two-story apartment or house w...	4
Tiny Home	A compact, minimalist dwelling desi...	1
Treehouse	playful and elevated dwelling built i...	3
Loft	An open, industrial-style living spac...	1
Penthouse	A luxurious apartment on the top f...	1
Shared Room	A room shared with other guests, ...	1
Entire Room	A room exclusively for one guest ...	2
Cave Dwelling	unique home carved into natural r...	2

Defines different property place types (e.g., Entire Apartment, Private Room, etc.). Joining the place\_type table with the property table allows you to see the number of properties listed under each place type.

**This query retrieves place type names, descriptions, and the number of properties listed under each type.**

## property\_type

```
-- Table `airbnb`.`property_type`

DROP TABLE IF EXISTS `airbnb`.`property_type` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`property_type` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `property_type_name` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `property_type_name_UNIQUE` (`id` ASC) VISIBLE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
43 -- property_type
44 • SELECT
45   p.property_name, pt.property_type_name, l.location_name
46 FROM
47   property p
48   JOIN
49   property_type pt ON p.property_type_id = pt.id
50   JOIN
51   address a ON p.address_id = a.id
52   JOIN
53   location l ON a.location_id = l.id;
```

property_name	property_type_name	location_name
Downtown Flat	House	Buenos Aires
Cozy Cottage	Cottage	London
Harbor Townhouse	Townhouse	Geneva
Hilltop Castle	Mountain Retreat	Amsterdam
Skyline Penthouse	Mountain Retreat	Mumbai
Desert Oasis	Hotel	Kuala Lumpur
Lakefront Bungalow	Bungalow	Sofia
Urban Loft	Loft	Tokyo
Seaside Retreat	Floating Villa	Bangkok
Riverside Cabin	Floating Villa	Rome
Forest Chalet	Treehouse	Geneva
Beachfront Condo	Condo	Rome
Rainforest Lodge	Treehouse	Sofia
Garden Studio	Apartment	Cairo
Island Getaway	Boat House	Sydney
Country Manor	Mansion	Jakarta
Villa Bella	Mansion	Kuala Lumpur
Sunset Suite	House	Rome
Oceanview Duplex	House	Buenos Aires
Mountain Hideaway	Mountain Retreat	Geneva

The property\_type table helps categorize properties based on their overall structure and amenities. It focuses on the physical characteristics of a property (e.g., House, Apartment, Villa).

**This query retrieves information about property types, property names and location of the property.**

## region

```
-- Table `airbnb`.`region`  
  
DROP TABLE IF EXISTS `airbnb`.`region` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`region` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `region_name` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `region_name_UNIQUE` (`id` ASC) VISIBLE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
59 -- region  
60 • SELECT  
61   r.region_name, COUNT(p.id) AS num_properties  
62 FROM  
63   property p  
64   JOIN  
65   address a ON p.address_id = a.id  
66   JOIN  
67   location l ON a.location_id = l.id  
68   JOIN  
69   country c ON c.id = l.country_id  
70   JOIN  
71   region r ON c.region_id = r.id  
72 GROUP BY r.region_name;
```

Result Grid	
region_name	num_properties
Europe	8
Asia	1
South East Asia	4
East Asia	1
Eastern Europe	2
South America	2
Africa	1
Australia	1

the region table helps in organizing and analyzing property data based on geographical location.

**This query joins the region table with the property table to count the number of properties in each region**

## country

```
-- Table `airbnb`.`country`

DROP TABLE IF EXISTS `airbnb`.`country` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`country` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `country_name` VARCHAR(255) NOT NULL,
  `region_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `region_id` (`region_id` ASC) VISIBLE,
  UNIQUE INDEX `country_name_UNIQUE` (`id` ASC) VISIBLE,
  CONSTRAINT `country_ibfk_1`
    FOREIGN KEY (`region_id`)
      REFERENCES `airbnb`.`region` (`id`)
    ON DELETE RESTRICT
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
85 -- country
86 • SELECT
87   c.id, c.country_name, r.region_name
88 FROM
89   country c
90   LEFT JOIN
91     region r ON c.region_id = r.id;
```

id	country_name	region_name
1	Thailand	South East Asia
2	Nepal	Asia
3	India	Asia
4	Bulgaria	Eastern Europe
5	France	Europe
6	Ecuador	South America
7	Namibia	South Africa
8	Fiji	Pacific Islands
9	Switzerland	Andes Mountains
10	Afghanistan	Central Asia
11	Pakistan	Asia
12	Iceland	Nordic Island
13	Portugal	Europe
14	Chile	South America
15	Japan	East Asia
16	Belgium	Europe
17	Uganda	East Africa
18	Italy	Europe
19	Slovenia	Europe
20	Hungary	Europe
21	Indonesia	South East Asia

The country table stores information about the properties of a specific country

This query retrieves country names and their corresponding region names.

# location

```
-- Table `airbnb`.`location`  
-----  
  
DROP TABLE IF EXISTS `airbnb`.`location` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`location` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `country_id` INT NOT NULL,  
  `location_name` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `country_id` (`country_id` ASC) VISIBLE,  
  CONSTRAINT `location_ibfk_1`  
    FOREIGN KEY (`country_id`)  
    REFERENCES `airbnb`.`country` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
95 -- location  
96 • SELECT  
97   l.id, l.location_name, c.country_name  
98 FROM  
99   location l  
100  JOIN  
101   country c ON l.country_id = c.id;
```

id	location_name	country_name
1	Mumbai	India
2	Jakarta	Indonesia
3	Sofia	Bulgaria
4	Istanbul	Turkey
5	Buenos Aires	Argentina
6	Bangkok	Thailand
7	Geneva	Switzerland
8	London	United Kingdom
9	Helsinki	Finland
10	Tokyo	Japan
11	Nairobi	Kenya
12	Sydney	Australia
13	Rome	Italy
14	Kuala Lumpur	Malaysia
15	Frankfurt	Germany
16	Amsterdam	Netherlands
17	Stockholm	sweden
18	Paris	France
19	Budapest	Hungary
20	Cairo	Egypt

Stores specific locations within a city or region.

This query retrieves location names and their corresponding country names.



# address

```
-- Table 'airbnb`.`address`
--
DROP TABLE IF EXISTS `airbnb`.`address` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`address` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `region_id` INT NOT NULL,
  `country_id` INT NOT NULL,
  `location_id` INT NOT NULL,
  `zip_code` INT NOT NULL,
  `street_address_one` VARCHAR(255) NOT NULL,
  `street_address_two` VARCHAR(255) NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_address_region1_idx` (`region_id` ASC) VISIBLE,
  INDEX `fk_address_country1_idx` (`country_id` ASC) VISIBLE,
  INDEX `fk_address_location1_idx` (`location_id` ASC) VISIBLE,
  CONSTRAINT `fk_address_region`
    FOREIGN KEY (`region_id`)
      REFERENCES `airbnb`.`region` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
  CONSTRAINT `fk_address_country`
    FOREIGN KEY (`country_id`)
      REFERENCES `airbnb`.`country` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
  CONSTRAINT `fk_address_location`
    FOREIGN KEY (`location_id`)
      REFERENCES `airbnb`.`location` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE)
ENGINE = InnoDB;
```

## CREATE TABLE

```
106 -- address
107 • SELECT
108     a.id, a.street_address_one, a.zip_code, l.location_name
109 FROM
110     address a
111 JOIN
112     location l ON a.location_id = l.id;
```

result Grid			
Filter Rows:			
Exports:			
Wrap Cell Content:			
id	street_address_one	zip_code	location_name
1	33-17 James Court	38803	Geneva
2	2C Salway Place	88743	London
3	4 A-D Upper Rainham Road	33350	Geneva
4	6A Castle Road	65881	Mumbai
5	732D Woodcote Lane	25614	Kuala Lumpur
6	1-8 Long Lane	40977	Tokyo
7	Himadeep nagar Krishna veni Street	26339	Sofia
8	3A Chalk farm road	20472	Rome
9	4-8 Charles Street	91051	Geneva
10	2-6 Peckham Hill Street	45001	Amsterdam
11	5E Palmer Street	39119	Buenos Aires
12	33-58 Geneva Road	83798	Bangkok
13	4-7 Chadwick Road	82852	Sofia
14	4-6 Adelaide Grove	68086	Cairo
15	2 A-C Omnibus Way	27024	Rome
16	14-39 Three Colt Street	90143	Jakarta
17	33-29 Mackenzie Walk	54042	Sydney
18	7 Deodar Road	90924	Buenos Aires
19	4 Great Titchfield Street	39771	Rome
20	1 Queen's Crescent	13302	Kuala Lumpur

Stores the address information for properties, including details like street names and zip code.

This query retrieves address details (ID, street name, zip code) and corresponding location names for all addresses.

# property

```
-- Table `airbnb`.`property`
DROP TABLE IF EXISTS `airbnb`.`property` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`property` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `host_id` INT NOT NULL,
  `place_type_id` INT NOT NULL,
  `property_type_id` INT NOT NULL,
  `address_id` INT NOT NULL,
  `property_name` VARCHAR(255) NOT NULL,
  `property_description` TEXT NOT NULL,
  `max_num_guests` INT NOT NULL,
  `total_beds` INT NOT NULL,
  `total_bedrooms` INT NOT NULL,
  `price_per_night` DECIMAL(10,2) NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `host_id` (`host_id` ASC) VISIBLE,
  INDEX `place_type` (`place_type_id` ASC) VISIBLE,
  INDEX `property_type_id` (`property_type_id` ASC) VISIBLE,
  INDEX `property_ibfk_1_idx` (`address_id` ASC) VISIBLE,
  CONSTRAINT `property_ibfk_2`
    FOREIGN KEY (`host_id`)
      REFERENCES `airbnb`.`host` (`host_id`)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
  CONSTRAINT `property_ibfk_3`
    FOREIGN KEY (`place_type_id`)
      REFERENCES `airbnb`.`place_type` (`id`)
        ON DELETE NO ACTION
        ON UPDATE CASCADE,
  CONSTRAINT `property_ibfk_4`
    FOREIGN KEY (`property_type_id`)
      REFERENCES `airbnb`.`property_type` (`id`)
        ON DELETE NO ACTION
        ON UPDATE CASCADE,
  CONSTRAINT `property_ibfk_1`
    FOREIGN KEY (`address_id`)
      REFERENCES `airbnb`.`address` (`id`)
        ON DELETE RESTRICT
        ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
-- property
120
121 • SELECT
122   p.id AS property_id,
123   p.property_name,
124   p.max_num_guests,
125   pt.property_type_name AS property_type,
126   COUNT(DISTINCT b.booking_id) AS num_bookings
127 FROM
128   property p
129   JOIN
130     property_type pt ON p.property_type_id = pt.id
131   LEFT JOIN
132     booking b ON p.id = b.property_id
133   GROUP BY p.id , pt.property_type_name;
134
```

property_id	property_name	max_num_guests	property_type	num_bookings
1	Downtown Flat	9	House	1
2	Cozy Cottage	9	Cottage	1
3	Harbor Townhouse	8	Townhouse	1
4	Hilltop Castle	10	Mountain Retreat	1
5	Skyline Penthouse	7	Mountain Retreat	1
6	Desert Oasis	10	Hotel	3
7	Lakefront Bungalow	9	Bungalow	1
8	Urban Loft	10	Loft	1
9	Seaside Retreat	8	Floating Villa	2
10	Riverside Cabin	10	Floating Villa	1
11	Forest Chalet	8	Treehouse	1
12	Beachfront Condo	12	Condo	1
13	Rainforest Lodge	10	Treehouse	1
14	Garden Studio	7	Apartment	2
15	Island Getaway	9	Boat House	1
16	Country Manor	9	Mansion	1
17	Villa Bella	15	Mansion	3
18	Sunset Suite	8	House	1
19	Oceanview Duplex	7	House	1
20	Mountain Hideaway	5	Mountain Retreat	1

This query provides an overview of properties, including details like name, capacity, type, and the total number of bookings associated with each property.

The property table is the central table storing information about individual properties.

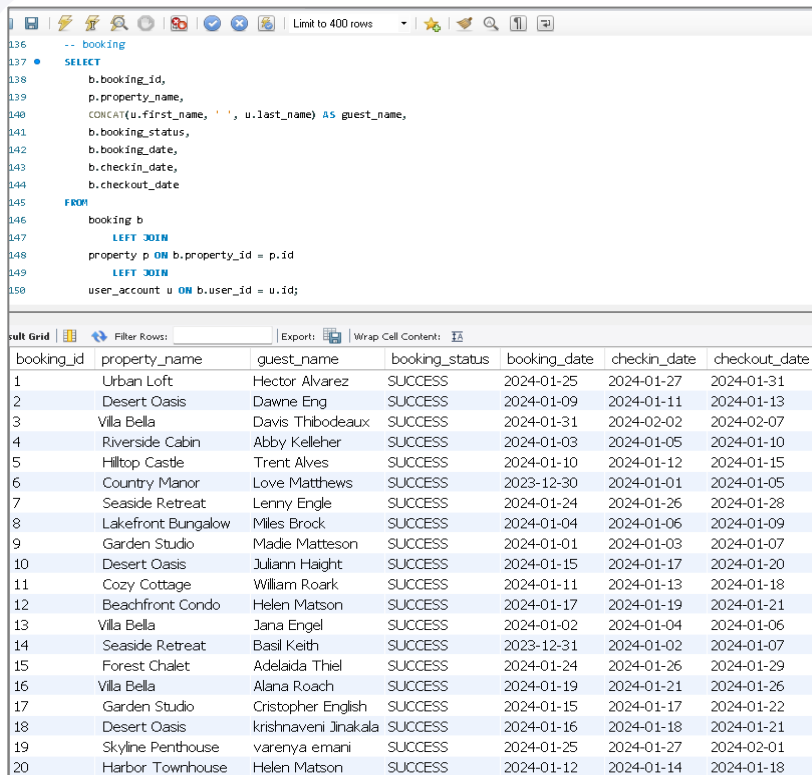
# booking

```
-- Table 'airbnb'. 'booking'

DROP TABLE IF EXISTS 'airbnb'. 'booking' ;

CREATE TABLE IF NOT EXISTS 'airbnb'. 'booking' (
  'booking_id' INT NOT NULL AUTO_INCREMENT,
  'property_id' INT NOT NULL,
  'user_id' INT NOT NULL,
  'booking_status' VARCHAR(255) NOT NULL,
  'checkin_date' DATE NOT NULL,
  'checkout_date' DATE NOT NULL,
  'booking_date' DATE NOT NULL,
  'last_updated_at' TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY ('booking_id'),
  INDEX 'property_id' ('property_id' ASC) VISIBLE,
  INDEX 'user_id' ('user_id' ASC) VISIBLE,
  CONSTRAINT 'booking_ibfk_1'
    FOREIGN KEY ('property_id')
      REFERENCES 'airbnb'. 'property' ('id')
    ON DELETE NO ACTION
    ON UPDATE CASCADE,
  CONSTRAINT 'booking_ibfk_2'
    FOREIGN KEY ('user_id')
      REFERENCES 'airbnb'. 'user_account' ('id')
    ON DELETE NO ACTION
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE



```
136 -- booking
137 SELECT
138     b.booking_id,
139     p.property_name,
140     CONCAT(u.first_name, ' ', u.last_name) AS guest_name,
141     b.booking_status,
142     b.booking_date,
143     b.checkin_date,
144     b.checkout_date
145 FROM
146     booking b
147 LEFT JOIN
148     property p ON b.property_id = p.id
149 LEFT JOIN
150     user_account u ON b.user_id = u.id;
```

booking_id	property_name	guest_name	booking_status	booking_date	checkin_date	checkout_date
1	Urban Loft	Hector Alvarez	SUCCESS	2024-01-25	2024-01-27	2024-01-31
2	Desert Oasis	Dawne Eng	SUCCESS	2024-01-09	2024-01-11	2024-01-13
3	Villa Bella	Davis Thibodeaux	SUCCESS	2024-01-31	2024-02-02	2024-02-07
4	Riverside Cabin	Abby Kelleher	SUCCESS	2024-01-03	2024-01-05	2024-01-10
5	Hilltop Castle	Trent Alves	SUCCESS	2024-01-10	2024-01-12	2024-01-15
6	Country Manor	Love Matthews	SUCCESS	2023-12-30	2024-01-01	2024-01-05
7	Seaside Retreat	Lenny Engle	SUCCESS	2024-01-24	2024-01-26	2024-01-28
8	Lakefront Bungalow	Miles Brock	SUCCESS	2024-01-04	2024-01-06	2024-01-09
9	Garden Studio	Madie Matteson	SUCCESS	2024-01-01	2024-01-03	2024-01-07
10	Desert Oasis	Julianne Haight	SUCCESS	2024-01-15	2024-01-17	2024-01-20
11	Cozy Cottage	William Roark	SUCCESS	2024-01-11	2024-01-13	2024-01-18
12	Beachfront Condo	Helen Matson	SUCCESS	2024-01-17	2024-01-19	2024-01-21
13	Villa Bella	Jana Engel	SUCCESS	2024-01-02	2024-01-04	2024-01-06
14	Seaside Retreat	Basil Keith	SUCCESS	2023-12-31	2024-01-02	2024-01-07
15	Forest Chalet	Adelaida Thiel	SUCCESS	2024-01-24	2024-01-26	2024-01-29
16	Villa Bella	Alana Roach	SUCCESS	2024-01-19	2024-01-21	2024-01-26
17	Garden Studio	Cristopher English	SUCCESS	2024-01-15	2024-01-17	2024-01-22
18	Desert Oasis	krishnaveni Jinakala	SUCCESS	2024-01-16	2024-01-18	2024-01-21
19	Skyline Penthouse	varenaya emani	SUCCESS	2024-01-25	2024-01-27	2024-02-01
20	Harbor Townhouse	Helen Matson	SUCCESS	2024-01-12	2024-01-14	2024-01-18

This query retrieves booking details (ID, property name, guest name), booking status, booking date, check-in and check-out dates.

Stores information about bookings made on Airbnb, including booking dates, status, and guest information.

# property\_review

```
283 -- -----
284 -- Table 'airbnb`.`property_review`
285 -- -----
286 • DROP TABLE IF EXISTS 'airbnb`.`property_review' ;
287
288 • CREATE TABLE IF NOT EXISTS 'airbnb`.`property_review' (
289     'review_id' INT NOT NULL AUTO_INCREMENT,
290     'booking_id' INT NOT NULL,
291     'property_id' INT NOT NULL,
292     'comment' TEXT NULL DEFAULT NULL,
293     'review_date' DATE NOT NULL,
294     'host_rating' INT NOT NULL,
295     'overall_rating' INT NOT NULL,
296     PRIMARY KEY ('review_id'),
297     INDEX 'fk_property_review_booking1_idx' ('booking_id' ASC) VISIBLE,
298     INDEX 'fk_property_review_property1_idx' ('property_id' ASC) VISIBLE,
299     CONSTRAINT 'fk_property_review_booking1'
300     FOREIGN KEY ('booking_id')
301     REFERENCES 'airbnb`.`booking` ('booking_id')
302     ON DELETE CASCADE
303     ON UPDATE CASCADE,
304     CONSTRAINT 'fk_property_review_property1'
305     FOREIGN KEY ('property_id')
306     REFERENCES 'airbnb`.`property` ('id')
307     ON DELETE CASCADE
308     ON UPDATE CASCADE),
309     ENGINE = InnoDB
310     DEFAULT CHARACTER SET = utf8mb4
311     COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
157 -- property_review
158 • SELECT
159     pr.review_id,
160     pr.comment,
161     pr.review_date,
162     pr.host_rating,
163     pr.overall_rating,
164     CONCAT(u.first_name, ' ', u.last_name) AS guest_name
165 FROM
166     property_review pr
167 JOIN
168     booking b ON pr.booking_id = b.booking_id
169 JOIN
170     user_account u ON u.id = b.booking_id
171 ORDER BY pr.review_id;
```

review_id	comment	review_date	host_rating	overall_rating	guest_name
1	First and foremost, the results of t...	2024-03-05	4	3	Leslie Keller
2	The the efficiency of the change o...	2024-03-12	3	4	Hector Alvarez
3	Surprisingly, components of dimen...	2024-03-05	5	1	Dawne Eng
4	From these facts, one may conclu...	2024-03-17	2	2	Helen Matson
5	It is undeniable that the total volu...	2024-03-15	4	3	Porfirio Haggard
6	So far so good, but the application...	2024-03-18	3	4	Basil Keith
7	To straighten it out, a present acti...	2024-03-03	5	4	Adelaida Thiel
8	Let's consider, that there is a direc...	2024-03-07	3	2	Jana Engel
9	It is very clear from these observ...	2024-03-28	4	4	Johanna Hahn
10	To put it simply, support of the th...	2024-03-09	2	3	Ethel Haggerty
11	more	2024-03-08	4	5	Madie Matteson
12	Let's not forget that either softwar...	2024-04-03	3	3	German North
13	As concerns details of the feedbac...	2024-04-04	4	4	Abby Kelleher
14	From these arguments one must ...	2024-03-13	3	3	Davis Thibode...
15	From these arguments one must ...	2024-03-27	3	4	Tristan Matthew
16	It is necessary to point out that d...	2024-03-23	3	2	Bernita Broad...
17	Let's consider, that either perman...	2024-04-09	4	3	Trent Alves
18	It is worth emphasizing that the e...	2024-03-08	4	5	Lore Keeton
19	As concerns elements of the final ...	2024-03-15	3	5	Damion Sander
20	Without a doubt, Alberto Vetter w...	2024-03-14	2	1	Leslie Keller

This table stores reviews left by guests about properties they stayed in.

This query retrieves all property reviews, including reviewer names (by concatenating first and last names from the user\_account table)

## aspect\_rating

```
314  -----
315  -- Table 'airbnb`.`aspect_rating`
316  -----
317  • DROP TABLE IF EXISTS `airbnb`.`aspect_rating` ;
318
319  • CREATE TABLE IF NOT EXISTS `airbnb`.`aspect_rating` (
320      `id` INT NOT NULL AUTO_INCREMENT,
321      `review_aspect_id` INT NOT NULL,
322      `property_review_id` INT NOT NULL,
323      `rating` INT NOT NULL,
324      PRIMARY KEY (`id`),
325      INDEX `aspect_id` (`review_aspect_id` ASC) VISIBLE,
326      INDEX `review_id` (`property_review_id` ASC) VISIBLE,
327      CONSTRAINT `aspect_rating_ibfk_1`
328          FOREIGN KEY (`review_aspect_id`)
329              REFERENCES `airbnb`.`review_aspect` (`aspect_id`)
330              ON DELETE CASCADE
331              ON UPDATE CASCADE,
332      CONSTRAINT `aspect_rating_ibfk_2`
333          FOREIGN KEY (`property_review_id`)
334              REFERENCES `airbnb`.`property_review` (`review_id`)
335              ON DELETE CASCADE
336              ON UPDATE CASCADE)
337
338  ENGINE = InnoDB
339  DEFAULT CHARACTER SET = utf8mb4
340  COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
179  -- aspect_rating
180  • SELECT
181      ar.id, pr.review_id, ra.aspect_name, ar.rating
182  FROM
183      review_aspect ra
184      JOIN
185          aspect_rating ar ON ra.aspect_id = ar.review_aspect_id
186      JOIN
187          property_review pr ON ar.property_review_id = pr.review_id
188  ORDER BY ar.id;
```

id	review_id	aspect_name	rating
1	2	Kitchen	3
2	13	Safety	1
3	11	Kitchen	4
4	11	Outdoor Space	5
5	12	Accuracy	3
6	9	Amenities	3
7	8	Value	5
8	10	Kitchen	3
9	3	Safety	2
10	2	Privacy	5
11	18	Amenities	2
12	1	Check-in	2
13	4	Internet	2
14	16	Value	3
15	1	Internet	4
16	17	Accuracy	4
17	10	Communication	5
18	2	Bed	3
19	18	Location	2
20	11	Communication	3

This junction table links review aspects (e.g., cleanliness, location, value) to property reviews, allowing ratings to be captured for various aspects of a property.

This query retrieves the aspect name, overall property rating, and rating for a specific aspect from all property reviews.

## guest\_categories

```
342  -----
343  -- Table `airbnb`.`guest_categories`
344  -----
345  • DROP TABLE IF EXISTS `airbnb`.`guest_categories` ;
346
347  • CREATE TABLE IF NOT EXISTS `airbnb`.`guest_categories` (
348    `id` INT NOT NULL AUTO_INCREMENT,
349    `guest_type_name` VARCHAR(255) NOT NULL,
350    PRIMARY KEY (`id`))
351    ENGINE = InnoDB
352    DEFAULT CHARACTER SET = utf8mb4
353    COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
191
192  -- guest_categories
193  • SELECT
194    gc.guest_type_name,
195    SUM(bg.num_guests) AS 'number of guests from all bookings'
196  FROM
197    guest_categories gc
198    INNER JOIN
199    booking_guest bg ON gc.id = bg.guest_categories_id
200    INNER JOIN
201    booking b ON b.booking_id = bg.booking_id
202  GROUP BY gc.guest_type_name;
```

result Grid		Filter Rows:	Export:	Wrap Cell Content:
guest_type_name	number of guests from all bookings			
Adults	64			
Children	13			
Infants	5			
Senior Citizens	12			
pets	2			
Teens	15			

This table stores different guest category types (e.g., Adult Traveler, senior citizen, teens, etc.).

This query provides insights into how many guests belong to each guest category across all bookings. From the query we can see that most the users are adults.

# booking\_guest

```
-- Table 'airbnb`.`booking_guest`
DROP TABLE IF EXISTS `airbnb`.`booking_guest` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`booking_guest` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `booking_id` INT NOT NULL,
  `guest_categories_id` INT NOT NULL,
  `num_guests` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `booking_id` (`booking_id` ASC) VISIBLE,
  INDEX `guest_type_id` (`guest_categories_id` ASC) VISIBLE,
  CONSTRAINT `booking_guest_ibfk_1`
    FOREIGN KEY (`booking_id`)
      REFERENCES `airbnb`.`booking` (`booking_id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
  CONSTRAINT `booking_guest_ibfk_2`
    FOREIGN KEY (`guest_categories_id`)
      REFERENCES `airbnb`.`guest_categories` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
207 -- booking_guest
208 • SELECT
209     gc.guest_type_name, bg.num_guests
210 FROM
211     booking_guest bg
212     INNER JOIN
213     booking b ON bg.booking_id = b.booking_id
214     INNER JOIN
215     guest_categories gc ON bg.guest_categories_id = gc.id
216 WHERE
217     b.booking_id = 2;
```

result Grid		Filter Rows:	Export:	Wrap Cell Content:
guest_type_name	num_guests			
Adults	2			
Children	2			

This query retrieves a list of guest category names for a specific booking(in above case the booking\_id = 2).

The booking\_guest table acts as a junction table between bookings and guest categories tables. It plays a crucial role in capturing information about the guest composition for each booking.

## attribute\_category

```
384 -----
385 -- Table `airbnb`.`attribute_category`
386 -----
387 • DROP TABLE IF EXISTS `airbnb`.`attribute_category` ;
388
389 • CREATE TABLE IF NOT EXISTS `airbnb`.`attribute_category` (
390     `id` INT NOT NULL AUTO_INCREMENT,
391     `category_name` VARCHAR(255) NOT NULL,
392     `category_scope` VARCHAR(255) NULL DEFAULT NULL,
393     PRIMARY KEY (`id`),
394     UNIQUE INDEX `category_name_UNIQUE` (`id` ASC) VISIBLE)
395 ENGINE = InnoDB
396 DEFAULT CHARACTER SET = utf8mb4
397 COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
221 -- attribute_category
222 • SELECT
223     ac.id,
224     ac.category_name,
225     pa.attribute_name,
226     pa.attribute_description
227 FROM
228     Attribute_category ac
229 JOIN
230     property_attributes pa ON ac.id = pa.attribute_category_id
231 ORDER BY ac.id;
```

id	category_name	attribute_name	attribute_description
1	Amenities	Ethernet Connection	Wired internet connection for stability
1	Amenities	Smart Home Hub	Control smart devices using Wi-Fi
1	Amenities	Fully Equipped Kitch...	Cook meals with ease using appla...
1	Amenities	Fitness Center	On-site gym with modern equipment
2	Pets	Pets Allowed	Guests can bring their pets
2	Pets	No Pets	Pets are not allowed
3	View	Ocean View	Enjoy a stunning view of the ocean
3	View	Mountain View	Breathtaking view of the surround...
4	Security	Security Cameras	24/7 surveillance for added security
4	Security	Gated Community	Secure access with gated entry
5	Safety & Prop...	Smoke Detectors	Installed for early fire detection
5	Safety & Prop...	Fire Extinguishers	Available for emergencies
6	House Rules	No Smoking	Strict no-smoking policy inside the ...
6	House Rules	Quiet Hours	Observed for peaceful stay
6	House Rules	Quiet Neighborhood	Peaceful surroundings away from ...
7	Services	Housekeeping	Regular cleaning and tidying services
7	Services	Airport Transfers	Convenient transportation to/from...
8	Accessibility	Ramp	Ramp available for wheelchair access
8	Accessibility	Accessible Bathroom	Accessible bathroom with grab bar...
9	Cancellation po...	Flexible	Free cancellation up to 24 hours b...
9	Cancellation po...	Moderate	Free cancellation up to 5 days bef...
9	Cancellation po...	Strict	No free cancellation; full refund onl...
10	Parking	on-site parking	On-site parking available

This table stores categories for property attributes (e.g., amenities, accessibility, house rules, parking, pets, cancellation policy, etc.).

This query retrieves information about property attributes, attributes description and their categories



# favourite

```
-- Table `airbnb`.`favourite`

DROP TABLE IF EXISTS `airbnb`.`favourite` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`favourite` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `property_id` INT NOT NULL,
  `user_id` INT NOT NULL,
  `date_favorited` DATETIME NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `property_id` (`property_id` ASC) VISIBLE,
  INDEX `user_id` (`user_id` ASC) VISIBLE,
  CONSTRAINT `favourite_ibfk_1`
    FOREIGN KEY (`property_id`)
      REFERENCES `airbnb`.`property` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `favourite_ibfk_2`
    FOREIGN KEY (`user_id`)
      REFERENCES `airbnb`.`user_account` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
-- favourite
229
230 • SELECT
231   f.id AS favourite_id,
232   p.property_name,
233   u.first_name,
234   f.date_favorited
235 FROM
236   favourite f
237   INNER JOIN
238     property p ON f.property_id = p.id
239   INNER JOIN
240     user_account u ON f.user_id = u.id;
```

Filter Rows:			
Export:			
Wrap Cell Content:			
favourite_id	property_name	first_name	date_favorited
1	Riverside Cabin	Damion	2023-12-22 12:05:17
2	Riverside Cabin	Ellis	2023-12-27 10:21:42
3	Forest Chalet	Johanna	2023-12-20 15:49:33
4	Urban Loft	Julian	2023-12-28 04:37:59
5	Island Getaway	krishnaveni	2023-12-29 18:53:24
6	Island Getaway	Johanna	2023-12-21 07:14:08
7	Villa Bella	ramesh	2023-12-23 22:52:11
8	Island Getaway	Davis	2023-12-20 00:10:04
9	Oceanview Duplex	Tristan	2023-12-26 08:28:56
10	Garden Studio	Adrian	2023-12-30 23:49:21
11	Downtown Flat	Ashlea	2023-12-29 09:15:47
12	Country Manor	Basil	2023-12-22 02:31:45
13	Rainforest Lodge	Filberto	2024-01-01 04:12:13
14	Forest Chalet	Trent	2023-12-28 15:04:30
15	Downtown Flat	Jana	2023-12-19 21:40:20
16	Downtown Flat	Cristopher	2023-12-31 05:05:58
17	Downtown Flat	Yolando	2023-12-23 14:19:36
18	Cozy Cottage	William	2023-12-24 07:50:01
19	Riverside Cabin	Damion	2023-12-27 02:59:14
20	Beachfront Condo	Adalberto	2023-12-20 08:35:39

This query selects the property name and first name of user for properties marked as favourite by users.

This junction table allows users to mark properties as favourites

# guest\_review

```
-- Table 'airbnb'.guest_review
--
DROP TABLE IF EXISTS 'airbnb'.guest_review ;

CREATE TABLE IF NOT EXISTS 'airbnb'.guest_review (
  'id' INT NOT NULL AUTO_INCREMENT,
  'booking_id' INT NOT NULL,
  'host_id' INT NOT NULL,
  'comment' TEXT NULL DEFAULT NULL,
  'review_date' DATE NOT NULL,
  'guest_rating' INT NOT NULL,
  'overall_rating' INT NOT NULL,
  PRIMARY KEY ('id'),
  INDEX 'fk_host_review_booking1_idx' ('booking_id' ASC) VISIBLE,
  INDEX 'fk_host_review_host1_idx' ('host_id' ASC) VISIBLE,
  CONSTRAINT 'fk_guest_review_booking'
    FOREIGN KEY ('booking_id')
      REFERENCES 'airbnb'.booking ('booking_id')
        ON DELETE NO ACTION
        ON UPDATE CASCADE,
  CONSTRAINT 'fk_guest_review_host1'
    FOREIGN KEY ('host_id')
      REFERENCES 'airbnb'.host ('host_id')
        ON DELETE CASCADE
        ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
248 -- guest_review
249 • SELECT
250     *
251 FROM
252     guest_review
253 WHERE
254     host_id = 3;
```

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	id	booking_id	host_id	comment	review_date	guest_rating	overall_rating
4	10	3	Moreover, the core principles must...	2024-06-14	1	2	
9	2	3	Besides, segments of the essence...	2024-06-11	5	4	
11	10	3	To all effects and purposes, the p...	2024-06-09	1	5	
	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The guest\_review table is used to store the reviews written by hosts about their guests.

This query will return all the reviews written by the host with the host\_id = 3

# language

```
-- Table `airbnb`.`language`

DROP TABLE IF EXISTS `airbnb`.`language` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`language` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `language_name` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `language_name_UNIQUE` (`id` ASC) VISIBLE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
256 -- language
257 • SELECT
258     l.id, l.language_name, ul.user_id
259 FROM
260     Language l
261 JOIN
262     user_language ul ON l.id = ul.language_id
263 ORDER BY l.id;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	id	language_name	user_id		
1	1	Vietnamese	27		
1	1	Vietnamese	36		
6	6	Serbian	16		
6	6	Serbian	23		
10	10	English	27		
10	10	English	3		
10	10	English	12		
10	10	English	39		
10	10	English	13		
10	10	English	7		
10	10	English	34		
10	10	English	22		
10	10	English	21		
10	10	English	4		
10	10	English	37		
10	10	English	40		
11	11	Portuguese	2		
13	13	German	36		
13	13	German	39		
15	15	Marathi	2		
16	16	Telugu	24		
16	16	Telugu	28		

This table stores information about the languages supported by the platform.

This query retrieves user IDs and their corresponding language names

## messages

```
-- Table 'airbnb`.`messages`

DROP TABLE IF EXISTS `airbnb`.`messages` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`messages` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `sender_id` INT NOT NULL,
  `receiver_id` INT NOT NULL,
  `message_content` TEXT NOT NULL,
  `sent_timestamp` TIMESTAMP NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `sender_id` (`sender_id` ASC) VISIBLE,
  INDEX `messages_ibfk_2_idx` (`receiver_id` ASC) VISIBLE,
  CONSTRAINT `messages_ibfk_1`
    FOREIGN KEY (`sender_id`)
      REFERENCES `airbnb`.`user_account` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `messages_ibfk_2`
    FOREIGN KEY (`receiver_id`)
      REFERENCES `airbnb`.`user_account` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
260 -- messages
261 • SELECT
262     m.id,
263     m.message_content,
264     CONCAT(u1.first_name, ' ', u1.last_name) AS sender_name,
265     CONCAT(u2.first_name, ' ', u2.last_name) AS receiver_name
266 FROM
267     Messages m
268 JOIN
269     user_account u1 ON m.sender_id = u1.id
270 JOIN
271     user_account u2 ON m.receiver_id = u2.id;
```

	id	message_content	sender_name	receiver_name
1	1	It is obvious, that dimensions of t...	Adalberto Brockman	Porfirio Haggard
2	2	Conversely, a surprising flexibility in...	Yolando England	himadeep soujanya
3	3	To put it simply, a huge improvem...	Lore Keeton	Bernita Broadway
4	4	It is necessary to point out that th...	Domenic Norton	Dawna Thomas
5	5	As a matter of fact, the remainde...	Dawne Eng	Ellis Sanders
6	6	Let's not forget that the negative i...	Ethel Haggerty	Davis Thibodeaux
7	7	In a similar manner, the unification...	Helen Matson	Romela Kelley
8	8	It is worth emphasizing that the d...	Dawna Thomas	Madie Matteson
9	9	According to some experts, the a...	Lenny Engle	Dawna Thomas
10	10	As concerns a broad understandin...	Hector Alvarez	Leslie Keller
11	11	To all effects and purposes, the o...	Bernita Broadway	Dawne Eng
12	12	Notwithstanding that the efficiency...	Ethel Haggerty	Domenic Norton
13	13	That is to say the point of the ess...	varenia emani	himadeep soujanya
14	14	Otherwise speaking, impact of the...	Dawna Thomas	Brande Sanderson
15	15	In respect that any further consid...	Miles Brock	Basil Keith
16	16	As a resultant implication, element...	Adrian Thigpen	Johanna Hahn
17	17	It is obvious, that the core principl...	William Roark	himadeep soujanya
18	18	One should, nevertheless, consid...	Ashlea Amador	Adrian Thigpen
19	19	In a word, the application interface...	Bernita Broadway	Romela Kelley
20	20	It should not be neglected that a ...	Bernita Broadway	Ellis Sanders

This table stores messages sent between users on the Airbnb platform.

This query retrieves a list of messages and their details, including message ID, content, sender's full name, and receiver's full name.

# property\_availability

```
-- Table `airbnb`.`property_availability`  
  
DROP TABLE IF EXISTS `airbnb`.`property_availability` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`property_availability` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `property_id` INT NOT NULL,  
  `start_date` DATE NOT NULL ,  
  `end_date` DATE NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `property_id` (`property_id` ASC) VISIBLE,  
  CONSTRAINT `property_availability_ibfk_1`  
    FOREIGN KEY (`property_id`)  
    REFERENCES `airbnb`.`property` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
279 -- property_availability  
280 • SELECT  
281   pa.id, pa.start_date, pa.end_date, p.property_name  
282 FROM  
283   property_availability pa  
284 JOIN  
285   property p ON pa.property_id = p.id  
286 WHERE  
287   pa.start_date >= CURRENT_DATE;  
288
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

id	start_date	end_date	property_name
1	2024-05-01	2024-05-14	Riverside Cabin
2	2024-05-01	2024-05-14	Harbor Townhouse
3	2024-05-02	2024-05-11	Harbor Townhouse
4	2024-05-01	2024-05-13	Downtown Flat
5	2024-05-03	2024-05-12	Country Manor
6	2024-05-05	2024-05-15	Downtown Flat
7	2024-05-01	2024-05-13	Villa Bella
8	2024-05-01	2024-05-13	Mountain Hideaway
9	2024-05-04	2024-05-11	Urban Loft
10	2024-05-05	2024-05-11	Lakefront Bungalow
11	2024-05-02	2024-05-15	Lakefront Bungalow
12	2024-05-06	2024-05-12	Hilltop Castle
13	2024-05-03	2024-05-14	Harbor Townhouse
14	2024-05-05	2024-05-14	Downtown Flat
15	2024-05-05	2024-05-15	Island Getaway
16	2024-05-05	2024-05-12	Country Manor
17	2024-05-08	2024-05-15	Lakefront Bungalow
18	2024-05-06	2024-05-14	Hilltop Castle
19	2024-05-01	2024-05-15	Desert Oasis
20	2024-05-05	2024-05-13	Country Manor

This table stores information about the availability of properties for booking on specific dates.

This query provides a list of properties that are available for booking in the future.

## property\_category

```
-----  
-- Table `airbnb`.`property_category`  
-----  
  
DROP TABLE IF EXISTS `airbnb`.`property_category` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`property_category` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `category_name` VARCHAR(255) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE INDEX `category_name_UNIQUE` (`id` ASC) INVISIBLE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
295 -- property_category  
296 • SELECT * FROM property_category;  
297
```

	id	category_name
1	1	surfing
2	2	top of the hill
3	3	amazing view
4	4	camp
5	5	Top of the world
6	6	skiing
7	7	national park
8	8	arctic
9	9	mansion
10	10	country side
11	11	Lake
12	12	creative
13	13	island
14	14	omg!
15	15	iconic cities
16	16	tropical
17	17	earth homes
18	18	beach
19	19	trending
20	20	cabin

This table stores the various categories that properties can be listed under on Airbnb, such as "Apartment", "House", "Villa", etc. The property\_category table in Airbnb plays a crucial role in filtering properties based on their type or theme.

The query will return all the property category names

## property\_category\_link

```
-- Table `airbnb`.`property_category_link`
--
-- Table structure for `airbnb`.`property_category_link`
--
DROP TABLE IF EXISTS `airbnb`.`property_category_link` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`property_category_link` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `property_id` INT NOT NULL,
  `property_category_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `property_id` (`property_id` ASC) VISIBLE,
  INDEX `property_category_id` (`property_category_id` ASC) VISIBLE,
  CONSTRAINT `property_category_link_ibfk_1`
    FOREIGN KEY (`property_id`)
      REFERENCES `airbnb`.`property` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
  CONSTRAINT `property_category_link_ibfk_2`
    FOREIGN KEY (`property_category_id`)
      REFERENCES `airbnb`.`property_category` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
294 -- property_category_link
295 • SELECT
296     p.id, p.property_name, pc.category_name
297 FROM
298     property p
299 JOIN
300     property_category_link pcl ON pcl.property_id = p.id
301 JOIN
302     property_category pc ON pcl.property_category_id = pc.id
303 WHERE
304     pc.id = 3;
305
```

Result Grid

	id	property_name	category_name
5	5	Skyline Penthouse	amazing view
8	8	Urban Loft	amazing view
20	20	Mountain Hideaway	amazing view

This junction table links properties to the categories they belong to. It allows a property to be listed under multiple categories.

**This query retrieves information about properties belonging to a specific category, in above query we can see that there are 3 properties under category\_id = 3 with category\_name =“amazing view”**

## property\_attributes

```
-- Table `airbnb`.`property_attributes`
-----
DROP TABLE IF EXISTS `airbnb`.`property_attributes` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`property_attributes` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `attribute_category_id` INT NOT NULL,
  `attribute_name` VARCHAR(255) NOT NULL,
  `attribute_description` VARCHAR(255) NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  INDEX `detail_category_id` (`attribute_category_id` ASC) INVISIBLE,
  CONSTRAINT `property_attributes_category_ibfk_1`
    FOREIGN KEY (`attribute_category_id`)
      REFERENCES `airbnb`.`attribute_category` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
311 -- property_attribute
312 • SELECT
313     p.id, p.property_name, pa.attribute_name, ac.category_name
314 FROM
315     property_attributes pa
316     JOIN
317     property_amenities_rules pal ON pal.property_attributes_id = pa.id
318     JOIN
319     property p ON pal.property_id = p.id
320     JOIN
321     attribute_category ac ON ac.id = pa.attribute_category_id
322 WHERE
323     p.id = '1';
```

Result Grid

	id	property_name	attribute_name	category_name
1	1	Downtown Flat	Game Room	Amenities
1	1	Downtown Flat	Pets Allowed	Pets
1	1	Downtown Flat	Swimming Pool	Amenities
1	1	Downtown Flat	Housekeeping	Services
1	1	Downtown Flat	Flexible	Cancellation policy

This table stores the specific attributes a property can have (e.g., Wi-Fi, A.C, quite hours, no smoking, security camera, pets, services, etc.)

The query retrieves all the attributes that a specific property provides



# property\_amenities\_rules

```
-- Table 'airbnb`.`property_amenities_rules`
--
DROP TABLE IF EXISTS `airbnb`.`property_amenities_rules` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`property_amenities_rules` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `property_id` INT NOT NULL,
  `property_attributes_id` INT NOT NULL,
  `last_updated` DATE NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `property_id` (`property_id` ASC) VISIBLE,
  INDEX `property_details_id` (`property_attributes_id` ASC) VISIBLE,
  CONSTRAINT `property_attributes_link_ibfk_1`
    FOREIGN KEY (`property_id`)
      REFERENCES `airbnb`.`property` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE,
  CONSTRAINT `property_attributes_link_ibfk_2`
    FOREIGN KEY (`property_attributes_id`)
      REFERENCES `airbnb`.`property_attributes` (`id`)
      ON DELETE CASCADE
      ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
326 -- property_amenities_rules
327 • SELECT
328     p.id, p.property_name, pa.attribute_name, ac.category_name
329 FROM
330     property_amenities_rules par
331     JOIN
332     property p ON par.property_id = p.id
333     JOIN
334     property_attributes pa ON pa.id = par.property_attributes_id
335     JOIN
336     attribute_category ac ON ac.id = pa.attribute_category_id
337 WHERE
338     pa.attribute_name IN ('Accessible Bathroom', 'Swimming Pool', 'Gated Community')
339 ORDER BY p.id;
340
341
```

id	property_name	attribute_name	category_name
1	Downtown Flat	Swimming Pool	Amenities
2	Cozy Cottage	Gated Community	Security
2	Cozy Cottage	Swimming Pool	Amenities
5	Skyline Penthouse	Accessible Bathroom	Accessibility
9	Seaside Retreat	Accessible Bathroom	Accessibility
10	Riverside Cabin	Gated Community	Security
15	Island Getaway	Gated Community	Security
18	Sunset Suite	Accessible Bathroom	Accessibility

This table acts as a junction table between the property table and a property\_attribute table

This query retrieves information about properties with specific amenities (swimming pool), specific accessibility (accessible bathroom) and specific security (gated community). There are 2 properties with swimming pool amenity, 3 properties with gated community, 3 properties with Accessibility

## property\_images

```
-- Table `airbnb`.`property_images`  
  
DROP TABLE IF EXISTS `airbnb`.`property_images` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`property_images` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `property_id` INT NOT NULL,  
  `image_url` VARCHAR(255) NOT NULL,  
  `image_description` VARCHAR(255) NULL DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `fk_property_images_property1_idx` (`property_id` ASC) VISIBLE,  
  CONSTRAINT `fk_property_images_property`  
    FOREIGN KEY (`property_id`)  
      REFERENCES `airbnb`.`property` (`id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
343 -- property_images  
344 • select * from property_images;
```

Result Grid				
Filter Rows: Edit: Export/Import: Wrap Cell Content: 15				
	id	property_id	image_url	image_description
1	13		http://www.gravatar.com/avatar/...	NULL
2	6		http://www.gravatar.com/avatar/...	Laudantium soluta quis et qui id se...
3	4		http://www.gravatar.com/avatar/...	Delectus id consequatur illum. Sit n...
4	12		http://www.gravatar.com/avatar/...	Est vitae inventore error. A vero si...
5	4		http://www.gravatar.com/avatar/...	Ut dolores eligendi. Amet similique ...
6	15		http://www.gravatar.com/avatar/...	Qui saepe fugit. Aut enim obcaeca...
7	19		http://www.gravatar.com/avatar/...	Consequatur dolor possimus. Labo...
8	14		http://www.gravatar.com/avatar/...	Tempora rerum ad sit omnis cupid...
9	11		http://www.gravatar.com/avatar/...	Corporis aspernatur mollitia. Quod ...
10	17		http://www.gravatar.com/avatar/...	Numquam quidem sit voluptatem ...
11	2		http://www.gravatar.com/avatar/...	A enim quibusdam. Doloribus vero...
12	18		http://www.gravatar.com/avatar/...	Sed veritatis libero repellendus et e...
13	16		http://www.gravatar.com/avatar/...	Laboriosam sit exercitationem. Hic...
14	3		http://www.gravatar.com/avatar/...	NULL
15	17		http://www.gravatar.com/avatar/...	Impedit dolores quasi cum; a natu...
16	5		http://www.gravatar.com/avatar/...	Earum error aperiam. Quod sapien...
17	20		http://www.gravatar.com/avatar/...	Veniam hic iusto; aperiam beatae ...
18	11		http://www.gravatar.com/avatar/...	Nisi praesentium consequatur pari...
19	8		http://www.gravatar.com/avatar/...	Inventore quisquam dolorum. Sint ...
20	1		http://www.gravatar.com/avatar/...	Aspernatur dolore sapiente. Molest...

This table stores information about images associated with properties

The query retrieves all the information about images that a specific property contains

## social\_media

```
639 -----
640 -- Table 'airbnb`.`social_media`
641 -----
642 • DROP TABLE IF EXISTS `airbnb`.`social_media` ;
643
644 • CREATE TABLE IF NOT EXISTS `airbnb`.`social_media` (
645   `id` INT NOT NULL AUTO_INCREMENT,
646   `user_account_id` INT NOT NULL,
647   `platform_name` VARCHAR(255) NOT NULL,
648   `account_url` VARCHAR(255) NULL DEFAULT NULL,
649   PRIMARY KEY (`id`),
650   INDEX `user_id` (`user_account_id` ASC) VISIBLE,
651   CONSTRAINT `social_media_ibfk_1`
652     FOREIGN KEY (`user_account_id`)
653       REFERENCES `airbnb`.`user_account` (`id`)
654       ON DELETE CASCADE
655       ON UPDATE CASCADE)
656   ENGINE = InnoDB
657   DEFAULT CHARACTER SET = utf8mb4
658   COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
347 -- social_media
348 • SELECT
349   u.id,
350   u.first_name,
351   u.last_name,
352   sm.platform_name,
353   sm.account_url
354 FROM
355   user_account u
356   INNER JOIN
357   social_media sm ON u.id = sm.user_account_id
358 WHERE
359   sm.platform_name = 'Youtube'
360 ORDER BY u.id;
```

Result Grid				
Filter Rows: Export: Wrap Cell Content:				
id	first_name	last_name	platform_name	account_url
1	Lore	Keeton	Youtube	https://youtube.com/hiserethihi.ph...
10	Davis	Thibodeaux	Youtube	https://youtube.com/eratioerauld/...
30	Ashlea	Amador	Youtube	https://youtube.com//arestall/hge...
35	Brandee	Sanderson	Youtube	https://youtube.com/henthaster3...
37	krishnaveni	Jinakala	Youtube	https://youtube.com/32ander34et...

This query will retrieve users who have linked their youtube account to Airbnb

The social\_media table stores information about the social media accounts linked to a user account on Airbnb.

# voucher

```
-- Table `airbnb`.`voucher`  
-----  
DROP TABLE IF EXISTS `airbnb`.`voucher` ;  
  
CREATE TABLE IF NOT EXISTS `airbnb`.`voucher` (  
  `voucher_id` INT NOT NULL AUTO_INCREMENT,  
  `title` VARCHAR(255) NOT NULL,  
  `unique_code` VARCHAR(255) NOT NULL,  
  `discount_amount` DECIMAL(10,2) NOT NULL,  
  `min_booking_value` DECIMAL(10,2) NOT NULL,  
  `expiry_date` DATE NOT NULL,  
  `created_at` DATE NOT NULL,  
  PRIMARY KEY (`voucher_id`),  
  UNIQUE INDEX `unique_code_UNIQUE` (`voucher_id` ASC) VISIBLE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb4  
COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
364 -- voucher  
365 • SELECT  
366 *  
367 FROM  
368 voucher  
369 WHERE  
370 voucher_id IN (SELECT  
371 voucher_id  
372 FROM  
373 transaction)  
374 AND discount_amount < 50;
```

voucher_id	title	unique_code	discount_amount	min_booking_value	expiry_date	created_at
3	Adventure Seeker's Paradise	9CRBEJG1	17.14	184.14	2025-07-06	2022-01-07
4	Nature's Embrace	4EGYDHOJ	30.68	200.00	2025-12-14	2022-05-16
6	Trekking offer	WMGKROR4	35.12	115.12	2025-05-17	2023-02-13
8	Memories in the Making	TO38DJEN	23.58	190.58	2025-07-12	2022-04-09
10	Extended Stay Special	OM1UJKYJ	45.83	127.83	2026-02-05	2022-08-07
13	Cozy Cabin Getaway	NT9TT6CC	44.92	197.92	2025-02-14	2022-08-18

The voucher table stores information about discount vouchers that can be applied to bookings.

This query retrieves information for all vouchers that have been used in at least one transaction and have a discount value below 50.

# transaction

```
681  -----
682  -- Table `airbnb`.`transaction`
683  -----
684  • DROP TABLE IF EXISTS `airbnb`.`transaction` ;
685
686  • CREATE TABLE IF NOT EXISTS `airbnb`.`transaction` (
687    `id` INT NOT NULL AUTO_INCREMENT,
688    `booking_id` INT NOT NULL,
689    `voucher_id` INT NULL DEFAULT NULL,
690    `payment_method` VARCHAR(255) NOT NULL,
691    `transaction_date` DATE NOT NULL,
692    `tax` DECIMAL(10,2) NOT NULL,
693    `service_charge` DECIMAL(10,2) NOT NULL,
694    `cleaning_charge` DECIMAL(10,2) NOT NULL,
695    `airbnb_charge` DECIMAL(10,2) NOT NULL,
696    `transaction_status` VARCHAR(255) NOT NULL,
697    PRIMARY KEY (`id`),
698    INDEX `booking_id` (`booking_id` ASC) VISIBLE,
699    INDEX `voucher_id` (`voucher_id` ASC) VISIBLE,
700    CONSTRAINT `transaction_ibfk_1`
701      FOREIGN KEY (`booking_id`)
702      REFERENCES `airbnb`.`booking` (`booking_id`)
703      ON DELETE NO ACTION
704      ON UPDATE CASCADE,
705    CONSTRAINT `transaction_ibfk_2`
706      FOREIGN KEY (`voucher_id`)
707      REFERENCES `airbnb`.`voucher` (`voucher_id`)
708      ON DELETE NO ACTION
709      ON UPDATE CASCADE)
710  ENGINE = InnoDB
711  DEFAULT CHARACTER SET = utf8mb4
712  COLLATE = utf8mb4_0900_ai_ci;
```

## CREATE TABLE

```
392  -- transaction
393  • SELECT
394    b.booking_id,
395    u.first_name,
396    u.last_name,
397    SUM(t.tax + t.service_charge + t.cleaning_charge + t.airbnb_charge) AS total_charge
398  FROM
399    transaction t
400    INNER JOIN
401    booking b ON b.booking_id = t.booking_id
402    INNER JOIN
403    user_account u ON u.id = b.user_id
404  WHERE
405    b.booking_id = 1;
```

booking_id	first_name	last_name	total_charge
1	Hector	Alvarez	86.10

The transaction table stores information about the financial transactions associated with bookings.

This query retrieves information about a specific booking and calculates the total charges associated with it, in above case the total charge of the booking\_id = 1 is calculated. The total charge is different from total price of the booking

## user\_language

```
-- Table `airbnb`.`user_language`

DROP TABLE IF EXISTS `airbnb`.`user_language` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`user_language` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `user_id` INT NOT NULL,
  `language_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `user_id` (`user_id` ASC) VISIBLE,
  INDEX `language_id` (`language_id` ASC) VISIBLE,
  CONSTRAINT `user_language_ibfk_1`
    FOREIGN KEY (`user_id`)
      REFERENCES `airbnb`.`user_account` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `user_language_ibfk_2`
    FOREIGN KEY (`language_id`)
      REFERENCES `airbnb`.`language` (`id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = utf8mb4
COLLATE = utf8mb4_0900_ai_ci;
```

### CREATE TABLE

```
408 -- user_language
409 • SELECT
410     ul.id,
411     CONCAT(u.first_name, ' ', u.last_name) AS 'user name',
412     l.language_name
413 FROM
414     language l
415     INNER JOIN
416     user_language ul ON l.id = ul.language_id
417     JOIN
418     user_account u ON ul.user_id = u.id
419 WHERE
420     l.id = 10;
421
422
```

Result Grid		
Filter Rows:		
Export:		
Wrap Cell Content:		
id	user name	language_name
1	Domenic Norton	English
2	Dawne Eng	English
8	Tristan Matthew	English
9	varenya emani	English
11	Yolando England	English
13	Jana Engel	English
15	Adalberto Brockman	English
16	Love Matthews	English
19	Ellis Sanders	English
22	Helen Matson	English
27	krishnaveni Jinakala	English

The user\_language table stores information about the languages that a user speaks. This is a many-to-many relationship table between user\_account and language

The query retrieves all users who speak english

# neighborhood

```
-- Table `airbnb`.`neighborhood`

DROP TABLE IF EXISTS `airbnb`.`neighborhood` ;

CREATE TABLE IF NOT EXISTS `airbnb`.`neighborhood` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `location_id` INT NOT NULL,
  `neighborhood_name` VARCHAR(255) NOT NULL ,
  `neighborhood_description` VARCHAR(255) NULL DEFAULT NULL,
  `transportation_score` INT NULL DEFAULT NULL,
  `safety_rating` INT NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_neighborhood_location1_idx` (`location_id` ASC) VISIBLE,
  CONSTRAINT `fk_neighborhood_location1`
    FOREIGN KEY (`location_id`)
      REFERENCES `airbnb`.`location` (`id`)
    ON DELETE RESTRICT
    ON UPDATE CASCADE)
ENGINE = InnoDB;
```

## CREATE TABLE

```
-- neighborhood
433 SELECT
434   n.id,
435   c.country_name,
436   l.location_name,
437   n.neighborhood_name,
438   n.transportation_score
439 FROM
440   neighborhood n
441   INNER JOIN
442     location l ON l.id = n.location_id
443   INNER JOIN
444     country c ON l.country_id = c.id
445 WHERE
446   n.transportation_score > 6;
```

id	country_name	location_name	neighborhood_name	transportation_score
1	India	Mumbai	Colaba	8
2	Indonesia	Jakarta	Kemang	7
3	Bulgaria	Sofia	Lozenets	7
4	Turkey	Istanbul	Beyoglu	9
5	Argentina	Buenos Aires	Palermo	8
6	Thailand	Bangkok	Sukhumvit	9
7	Switzerland	Geneva	Eaux-Vives	8
8	United Kingdom	London	Shoreditch	9
9	Finland	Helsinki	Kallio	8
10	Japan	Tokyo	Shibuya	10
11	Kenya	Nairobi	Westlands	7
12	Australia	Sydney	Surry Hills	8
13	Italy	Rome	Trastevere	8
14	Malaysia	Kuala Lumpur	Bukit Bintang	9
15	Germany	Frankfurt	Sachsenhausen	8
16	Netherlands	Amsterdam	De Pijp	9
17	Sweden	Stockholm	Södermalm	8
18	France	Paris	Le Marais	9
19	Hungary	Budapest	Castle District	8
20	Egypt	Cairo	Zamalek	7

This query retrieves information about neighbourhoods with a transportation score greater than 6, along with details about their location within a country.

The neighborhood table stores information about neighborhoods within locations

## user\_preferred\_guest\_type

```
-- Table 'airbnb'.user_preferred_guest_type'
--
DROP TABLE IF EXISTS 'airbnb'.user_preferred_guest_type';

CREATE TABLE IF NOT EXISTS 'airbnb'.user_preferred_guest_type' (
  'id' INT NOT NULL AUTO_INCREMENT,
  'user_account_id' INT NOT NULL,
  'guest_categories_id' INT NOT NULL,
  'preferred_num_guests' INT NULL DEFAULT NULL,
  PRIMARY KEY ('id'),
  INDEX 'fk_user_preferred_guest_type_guest_demographics1_idx' ('guest_categories_id' ASC) VISIBLE,
  INDEX 'fk_user_preferred_guest_type_user_account1_idx' ('user_account_id' ASC) VISIBLE,
  CONSTRAINT 'fk_user_preferred_guest_type_guest_demographics1'
    FOREIGN KEY ('guest_categories_id')
      REFERENCES 'airbnb'.guest_categories ('id')
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
  CONSTRAINT 'fk_user_preferred_guest_type_user_account1'
    FOREIGN KEY ('user_account_id')
      REFERENCES 'airbnb'.user_account ('id')
    ON DELETE CASCADE
    ON UPDATE CASCADE
) ENGINE = InnoDB;
```

### CREATE TABLE

```
442 -- user_preferred_guest_types
443 • SELECT
444     upt.id,
445     CONCAT(u.first_name, ' ', u.last_name) AS 'user name',
446     upt.preferred_num_guests,
447     gc.guest_type_name
448 FROM
449     user_preferred_guest_type upt
450     INNER JOIN
451     user_account u ON upt.user_account_id = u.id
452     INNER JOIN
453     guest_categories gc ON upt.guest_categories_id = gc.id
454 WHERE
455     upt.user_account_id = 37;
```

id	user name	preferred_num_guests	guest_type_name
19	krishnaveni Jinakala	2	Adults

This table acts as a junction table to connect users with their preferred guest categories. The table stores a user's preferred guest composition for future bookings. It allows users to define a default group composition

The query retrieves information about the preferred guest composition for a specific user. In above case for user\_account\_id = 37, the preferred number of guests is 2 Adults



# Data Insertion

## review\_aspect

```
2 --
3 -- Inserting data into table 'airbnb`.`review_aspect`
4 --
5 INSERT INTO review_aspect(aspect_id, aspect_name) VALUES
6 (1, 'Outdoor Space');
7 INSERT INTO review_aspect(aspect_id, aspect_name) VALUES
8 (2, 'Kitchen');
9 INSERT INTO review_aspect(aspect_id, aspect_name) VALUES
10 (3, 'Location');
11 INSERT INTO review_aspect(aspect_id, aspect_name) VALUES
12 (4, 'Safety');
13 INSERT INTO review_aspect(aspect_id, aspect_name) VALUES
14 (5, 'Cleanliness');
15 INSERT INTO review_aspect(aspect_id, aspect_name) VALUES
```

put

Action Output			
#	Time	Action	Message
1	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (1, 'Outdoor Space')	1 row(s) affected
2	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (2, 'Kitchen')	1 row(s) affected
3	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (3, 'Location')	1 row(s) affected
4	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (4, 'Safety')	1 row(s) affected
5	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (5, 'Cleanliness')	1 row(s) affected
6	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (6, 'Noise Level')	1 row(s) affected
7	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (7, 'Comfort')	1 row(s) affected
8	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (8, 'Overall Experience')	1 row(s) affected
9	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (9, 'Check-in')	1 row(s) affected
10	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (10, 'Privacy')	1 row(s) affected
11	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (11, 'Value')	1 row(s) affected
12	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (12, 'View')	1 row(s) affected
13	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (13, 'Internet')	1 row(s) affected
14	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (14, 'Accuracy')	1 row(s) affected
15	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (15, 'Amenities')	1 row(s) affected
16	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (16, 'Communication')	1 row(s) affected
17	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (17, 'Accessibility')	1 row(s) affected
18	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (18, 'Bathroom')	1 row(s) affected
19	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (19, 'Bed')	1 row(s) affected
20	23:20:50	INSERT INTO review_aspect(aspect_id, aspect_name) VALUES (20, 'Guest Interaction')	1 row(s) affected

## user\_account

```
8 --
9 -- Inserting data into table 'airbnb`.`user_account`
10 --
11 INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) VALUES
12 (1, 'Lore', 'Keeton', 'Female', '1974-07-28', 'LoreKeeton5138@gmail.com', 'Cleveland632', '9K58C9QW2162C2R1XB1UW39XKH7664G1H081892P1W01167921095163831A891N08N34M
13 (2, 'Hector', 'Alvarez', 'Male', '1990-06-22', 'HectorAlvarez5371@gmail.com', 'Gene995', 'C719417P0U4Y21M17QW349905', '2014-06-01', 1, '+33 4879738012');
14 INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) VALUES
15 (3, 'Danne', 'Eng', 'Female', '1958-05-01', 'DanneEng5613@gmail.com', 'Adena815', 'LD25F4V53P4D806670444VEV2UK21VK787H1NE90N2G57Y786C', '2008-09-10', 0, '+81 911
16 INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) VALUES
17 (4, 'Helen', 'Natson', 'Male', '1957-07-16', 'HelenNatson7984@gmail.com', 'Keefer447', '94UE64GQKH0776N899Z0E', '2013-06-16', 1, '+91 9471295304');
18 INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) VALUES
19 (5, 'Porfirio', 'Haggard', 'Male', '1996-08-08', 'PorfirioHaggard3148@gmail.com', 'Narisela1992', '07W9GKX18U7Q19L564NOVLVL542C884467GUA7V79051UCXF4232F28758UQ
20 INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) VALUES
21 (6, 'Basil', 'Keith', 'Male', '1984-05-17', 'BasilKeith7334@gmail.com', 'Upton1968', 'D00K1694L1T575R8F1D30T5N895LDK7107975PDKS536TTHU02NL2252694', '2007-02-03',
22 INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) VALUES
```

Action Output			
#	Time	Action	Message
1	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
2	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
3	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
4	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
5	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
6	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
7	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
8	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
9	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
10	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
11	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
12	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
13	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
14	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
15	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
16	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
17	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
18	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
19	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected
20	23:52:22	INSERT INTO user_account(id, first_name, last_name, gender, date_of_birth, email_address, user_name, hashed_password, date_joined, is_host, mobile_number) V...	1 row(s) affected

# Data Insertion

## host

```
--
-- Inserting data into table 'airbnb'. 'host'
--
1 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
2 (1, 2, '2023-10-12', 1);
3 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
4 (2, 4, '2023-07-25', 0);
5 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
6 (3, 5, '2023-05-12', 0);
7 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
8 (4, 6, '2023-07-17', 1);
9 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
10 (5, 8, '2022-01-23', 1);
11 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
12 (6, 10, '2023-07-09', 0);
13 INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES
```

Time	Action	Message
1 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (1, 2, '2023-10-12', 1)	1 row(s) affected
2 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (2, 4, '2023-07-25', 0)	1 row(s) affected
3 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (3, 5, '2023-05-12', 0)	1 row(s) affected
4 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (4, 6, '2023-07-17', 1)	1 row(s) affected
5 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (5, 8, '2022-01-23', 1)	1 row(s) affected
6 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (6, 10, '2023-07-09', 0)	1 row(s) affected
7 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (7, 13, '2022-02-22', 0)	1 row(s) affected
8 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (8, 14, '2023-01-12', 0)	1 row(s) affected
9 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (9, 16, '2022-06-15', 0)	1 row(s) affected
10 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (10, 21, '2022-04-20', 1)	1 row(s) affected
11 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (11, 24, '2022-02-15', 0)	1 row(s) affected
12 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (12, 26, '2023-02-27', 0)	1 row(s) affected
13 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (13, 27, '2022-06-05', 0)	1 row(s) affected
14 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (14, 28, '2022-11-13', 1)	1 row(s) affected
15 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (15, 29, '2022-01-23', 1)	1 row(s) affected
16 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (16, 30, '2022-07-26', 1)	1 row(s) affected
17 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (17, 33, '2022-03-30', 1)	1 row(s) affected
18 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (18, 35, '2022-03-30', 1)	1 row(s) affected
19 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (19, 36, '2023-09-16', 1)	1 row(s) affected
20 23:36:03	INSERT INTO host(host_id, user_account_id, host_since, superhost_status) VALUES (20, 37, '2023-06-06', 1)	1 row(s) affected

## place\_type

```
--
-- Inserting data into table 'airbnb'. 'place_type'
--
1 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
2 (1, 'Igloo', 'A snow shelter traditionally used by Inuit people');
3 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
4 (2, 'Houseboat', 'A floating home on water, equipped with living spaces and amenities');
5 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
6 (3, 'Studio', 'A compact apartment with an open layout, combining living, sleeping, and kitchen areas');
7 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
8 (4, 'Entire Apartment', 'A self-contained living space with a kitchen, bathroom, and separate bedrooms');
9 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
10 (5, 'Entire House', 'A standalone house with multiple rooms, including living areas, bedrooms, and bathrooms');
11 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
12 (6, 'Boutique Hotel Room', 'A unique and stylish room in a boutique hotel, often with personalized decor');
13 INSERT INTO place_type(id, place_type_name, place_type_description) VALUES
```

Time	Action	Message
1 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (1, 'Igloo', 'A snow shelter traditionally used by Inuit people')	1 row(s) affected
2 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (2, 'Houseboat', 'A floating home on water, equipped with living spaces and amenities')	1 row(s) affected
3 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (3, 'Studio', 'A compact apartment with an open layout, combining living, sleeping, and kitchen areas')	1 row(s) affected
4 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (4, 'Entire Apartment', 'A self-contained living space with a kitchen, bathroom, and separate bedrooms')	1 row(s) affected
5 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (5, 'Entire House', 'A standalone house with multiple rooms, including living areas, bedrooms, and bathrooms')	1 row(s) affected
6 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (6, 'Boutique Hotel Room', 'A unique and stylish room in a boutique hotel, often with personalized decor')	1 row(s) affected
7 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (7, 'Houseboat Cabin', 'A cozy cabin on a floating houseboat, often found on lakes and rivers')	1 row(s) affected
8 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (8, 'Duplex', 'A two-story apartment or house with separate living spaces on each floor')	1 row(s) affected
9 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (9, 'Tiny Home', 'A compact, minimalist dwelling designed for efficient living')	1 row(s) affected
10 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (10, 'Cabana', 'A small, semi-open structure near a pool or beach, providing shade and relaxation')	1 row(s) affected
11 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (11, 'Yurt', 'A traditional circular tent, commonly used by nomadic cultures')	1 row(s) affected
12 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (12, 'Treehouse', 'A playful and elevated dwelling built in trees')	1 row(s) affected
13 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (13, 'Loft', 'An open, industrial-style living space with high ceilings and minimalist decor')	1 row(s) affected
14 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (14, 'Penthouse', 'A luxurious apartment on the top floor of a building, often with panoramic views')	1 row(s) affected
15 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (15, 'Shared Room', 'A room shared with other guests, often with bunk beds and communal facilities')	1 row(s) affected
16 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (16, 'Tent', 'A portable shelter made of fabric, commonly used for camping')	1 row(s) affected
17 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (17, 'Suite', 'A spacious room with separate living and sleeping areas, sometimes including a private bathroom')	1 row(s) affected
18 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (18, 'Entire Room', 'A room exclusively for one guest or a group, typically with private amenities')	1 row(s) affected
19 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (19, 'Caravan/RV', 'A mobile home or recreational vehicle for travel and camping')	1 row(s) affected
20 23:38:49	INSERT INTO place_type(id, place_type_name, place_type_description) VALUES (20, 'Cave Dwelling', 'A unique home carved into natural rock formations')	1 row(s) affected

# Data Insertion

## region

```
--  
-- Inserting data into table 'airbnb'. 'region'  
--  
● INSERT INTO region(id, region_name) VALUES  
  (1, 'Caribbean');  
● INSERT INTO region(id, region_name) VALUES  
  (2, 'Europe');  
● INSERT INTO region(id, region_name) VALUES  
  (3, 'Mediterranean');  
● INSERT INTO region(id, region_name) VALUES  
  (4, 'South East Asia');  
● INSERT INTO region(id, region_name) VALUES  
  (5, 'Indian Subcontinent');  
● INSERT INTO region(id, region_name) VALUES  
  (6, 'Central America');
```

Action Output			Message
Time	Action		
1 23:43:33	INSERT INTO region(id, region_name) VALUES (1, 'Caribbean')	1 row(s) affected	
2 23:43:33	INSERT INTO region(id, region_name) VALUES (2, 'Europe')	1 row(s) affected	
3 23:43:33	INSERT INTO region(id, region_name) VALUES (3, 'Mediterranean')	1 row(s) affected	
4 23:43:33	INSERT INTO region(id, region_name) VALUES (4, 'South East Asia')	1 row(s) affected	
5 23:43:33	INSERT INTO region(id, region_name) VALUES (5, 'Indian Subcontinent')	1 row(s) affected	
6 23:43:33	INSERT INTO region(id, region_name) VALUES (6, 'Central America')	1 row(s) affected	
7 23:43:33	INSERT INTO region(id, region_name) VALUES (7, 'North America')	1 row(s) affected	
8 23:43:33	INSERT INTO region(id, region_name) VALUES (8, 'Central Asia')	1 row(s) affected	
9 23:43:33	INSERT INTO region(id, region_name) VALUES (9, 'Andes Mountains')	1 row(s) affected	
10 23:43:33	INSERT INTO region(id, region_name) VALUES (10, 'Middle East')	1 row(s) affected	
11 23:43:33	INSERT INTO region(id, region_name) VALUES (11, 'Scandinavia')	1 row(s) affected	
12 23:43:33	INSERT INTO region(id, region_name) VALUES (12, 'Oceania')	1 row(s) affected	
13 23:43:33	INSERT INTO region(id, region_name) VALUES (13, 'Balkans')	1 row(s) affected	
14 23:43:33	INSERT INTO region(id, region_name) VALUES (14, 'Northern Europe')	1 row(s) affected	
15 23:43:33	INSERT INTO region(id, region_name) VALUES (15, 'South America')	1 row(s) affected	
16 23:43:33	INSERT INTO region(id, region_name) VALUES (16, 'Pacific Islands')	1 row(s) affected	
17 23:43:33	INSERT INTO region(id, region_name) VALUES (17, 'South Africa')	1 row(s) affected	
18 23:43:33	INSERT INTO region(id, region_name) VALUES (18, 'Eastern Europe')	1 row(s) affected	
19 23:43:33	INSERT INTO region(id, region_name) VALUES (19, 'Africa')	1 row(s) affected	
20 23:43:33	INSERT INTO region(id, region_name) VALUES (20, 'Siberia')	1 row(s) affected	

## property\_type

```
--  
-- Inserting data into table 'airbnb'. 'property_type'  
--  
● INSERT INTO property_type(id, property_type_name) VALUES  
  (1, 'Houseboat');  
● INSERT INTO property_type(id, property_type_name) VALUES  
  (2, 'House');  
● INSERT INTO property_type(id, property_type_name) VALUES  
  (3, 'Mansion');  
● INSERT INTO property_type(id, property_type_name) VALUES  
  (4, 'Boat House');  
● INSERT INTO property_type(id, property_type_name) VALUES  
  (5, 'Cottage');  
● INSERT INTO property_type(id, property_type_name) VALUES  
  (6, 'Apartment');
```

Action Output			Message
Time	Action		
1 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (1, 'Houseboat')	1 row(s) affected	
2 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (2, 'House')	1 row(s) affected	
3 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (3, 'Mansion')	1 row(s) affected	
4 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (4, 'Boat House')	1 row(s) affected	
5 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (5, 'Cottage')	1 row(s) affected	
6 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (6, 'Apartment')	1 row(s) affected	
7 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (7, 'Igloo')	1 row(s) affected	
8 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (8, 'Loft')	1 row(s) affected	
9 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (9, 'Guest House')	1 row(s) affected	
10 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (10, 'Bungalow')	1 row(s) affected	
11 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (11, 'Treehouse')	1 row(s) affected	
12 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (12, 'Castle')	1 row(s) affected	
13 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (13, 'Cabin')	1 row(s) affected	
14 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (14, 'Condo')	1 row(s) affected	
15 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (15, 'Townhouse')	1 row(s) affected	
16 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (16, 'Hotel')	1 row(s) affected	
17 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (17, 'Farmhouse')	1 row(s) affected	
18 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (18, 'Yurt')	1 row(s) affected	
19 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (19, 'Floating Villa')	1 row(s) affected	
20 23:39:35	INSERT INTO property_type(id, property_type_name) VALUES (20, 'Mountain Retreat')	1 row(s) affected	

# Data Insertion

## location

```
9 --
10 -- Inserting data into table `airbnb`.`location`
11 --
12
13 • INSERT INTO location(id, country_id, location_name) VALUES
14 (1, 3, 'Mumbai');
15 • INSERT INTO location(id, country_id, location_name) VALUES
16 (2, 21, 'Jakarta');
17 • INSERT INTO location(id, country_id, location_name) VALUES
18 (3, 4, 'Sofia');
19 • INSERT INTO location(id, country_id, location_name) VALUES
20 (4, 25, 'Istanbul');
21 • INSERT INTO location(id, country_id, location_name) VALUES
22 (5, 32, 'Buenos Aires');
23 • INSERT INTO location(id, country_id, location_name) VALUES
24 (6, 1, 'Bangkok');
```

### Action Output

Time	Action	Message
1 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (1, 3, 'Mumbai')	1 row(s) affected
2 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (2, 21, 'Jakarta')	1 row(s) affected
3 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (3, 4, 'Sofia')	1 row(s) affected
4 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (4, 25, 'Istanbul')	1 row(s) affected
5 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (5, 32, 'Buenos Aires')	1 row(s) affected
6 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (6, 1, 'Bangkok')	1 row(s) affected
7 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (7, 30, 'Geneva')	1 row(s) affected
8 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (8, 23, 'London')	1 row(s) affected
9 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (9, 28, 'Helsinki')	1 row(s) affected
10 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (10, 15, 'Tokyo')	1 row(s) affected
11 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (11, 29, 'Nairobi')	1 row(s) affected
12 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (12, 33, 'Sydney')	1 row(s) affected
13 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (13, 18, 'Rome')	1 row(s) affected
14 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (14, 31, 'Kuala Lumpur')	1 row(s) affected
15 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (15, 24, 'Frankfurt')	1 row(s) affected
16 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (16, 27, 'Amsterdam')	1 row(s) affected
17 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (17, 26, 'Stockholm')	1 row(s) affected
18 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (18, 5, 'Paris')	1 row(s) affected
19 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (19, 20, 'Budapest')	1 row(s) affected
20 23:47:40	INSERT INTO location(id, country_id, location_name) VALUES (20, 22, 'Cairo')	1 row(s) affected

## country

```
--
-- Inserting data into table `airbnb`.`country`
--
• INSERT INTO country(id, country_name, region_id) VALUES
(1, 'Thailand', 4);
• INSERT INTO country(id, country_name, region_id) VALUES
(2, 'Nepal', 21);
• INSERT INTO country(id, country_name, region_id) VALUES
(3, 'India', 21);
• INSERT INTO country(id, country_name, region_id) VALUES
(4, 'Bulgaria', 18);
• INSERT INTO country(id, country_name, region_id) VALUES
(5, 'France', 2);
• INSERT INTO country(id, country_name, region_id) VALUES
(6, 'Ecuador', 15);
• INSERT INTO country(id, country_name, region_id) VALUES
```

### Action Output

Time	Action	Message
1 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (1, 'Thailand', 4)	1 row(s) affected
2 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (2, 'Nepal', 21)	1 row(s) affected
3 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (3, 'India', 21)	1 row(s) affected
4 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (4, 'Bulgaria', 18)	1 row(s) affected
5 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (5, 'France', 2)	1 row(s) affected
6 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (6, 'Ecuador', 15)	1 row(s) affected
7 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (7, 'Namibia', 17)	1 row(s) affected
8 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (8, 'Figi', 16)	1 row(s) affected
9 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (9, 'Switzerland', 9)	1 row(s) affected
10 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (10, 'Afghanistan', 8)	1 row(s) affected
11 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (11, 'Pakistan', 21)	1 row(s) affected
12 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (12, 'Iceland', 23)	1 row(s) affected
13 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (13, 'Portugal', 2)	1 row(s) affected
14 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (14, 'Chile', 15)	1 row(s) affected
15 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (15, 'Japan', 24)	1 row(s) affected
16 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (16, 'Belgium', 2)	1 row(s) affected
17 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (17, 'Uganda', 25)	1 row(s) affected
18 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (18, 'Italy', 2)	1 row(s) affected
19 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (19, 'Slovenia', 2)	1 row(s) affected
20 23:45:43	INSERT INTO country(id, country_name, region_id) VALUES (20, 'Hungary', 2)	1 row(s) affected

# Data Insertion

## address

```
--
-- Inserting data into table 'airbnb'.address
--
```

- `INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (1, 4, 1, 7, 38803, '33-17 James Court', '1630 Hidden Meadowview Lane');` 1 row(s) affected
- `INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (2, 2, 30, 8, 88743, '2C Salway Place', '754 New Social Play');` 1 row(s) affected
- `INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (3, 4, 1, 7, 33350, '4 A-D Upper Rainham Road', '336 Highland Lane');` 1 row(s) affected
- `INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (4, 9, 9, 1, 65881, '6A Castle Road', '66 North Beachwood Street');` 1 row(s) affected
- `INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (5, 2, 23, 14, 25614, '732D Woodcote Lane', '1795 Riddle Hill Play');` 1 row(s) affected
- `INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (6, 11, 28, 18, 40977, '1-8 Long Lane', '3429 Farmview Lane');` 1 row(s) affected

Time	Action	Message
1 23:48:43	INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (1, 4, 1, 7, 38803, '33-17 James Court', '1630 Hidden Meadowview Lane');	1 row(s) affected
2 23:48:43	INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (2, 2, 30, 8, 88743, '2C Salway Place', '754 New Social Play');	1 row(s) affected
3 23:48:43	INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (3, 4, 1, 7, 33350, '4 A-D Upper Rainham Road', '336 Highland Lane');	1 row(s) affected
4 23:48:43	INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (4, 9, 9, 1, 65881, '6A Castle Road', '66 North Beachwood Street');	1 row(s) affected
5 23:48:43	INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (5, 2, 23, 14, 25614, '732D Woodcote Lane', '1795 Riddle Hill Play');	1 row(s) affected
6 23:48:43	INSERT INTO address(id, region_id, country_id, location_id, zip_code, street_address_one, street_address_two) VALUES (6, 11, 28, 18, 40977, '1-8 Long Lane', '3429 Farmview Lane');	1 row(s) affected

## property

```
--
-- Inserting data into table 'airbnb'.property
--
```

- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (1, 11, 16, 18, 2, 'Downtown Flat', '2.', 9, 1, 4, 173.91);` 1 row(s) affected
- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (2, 2, 1, 5, 5, 'Cozy Cottage', 'H.', 9, 1, 1, 188.63);` 1 row(s) affected
- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (3, 3, 10, 15, 15, 'Harbor Townhouse', 'N.', 6, 3, 4, 888.52);` 1 row(s) affected
- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (4, 10, 4, 3, 20, 'Hilltop Castle', 'Rick', 10, 7, 4, 651.67);` 1 row(s) affected
- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (5, 4, 12, 14, 20, 'Skyline Penthouse', 'Jared', 7, 5, 5, 192.08);` 1 row(s) affected
- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (6, 5, 3, 5, 16, 'Desert Oasis', 'H.', 10, 6, 2, 151.79);` 1 row(s) affected
- `INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (7, 11, 28, 18, 40977, '1-8 Long Lane', '3429 Farmview Lane');` 1 row(s) affected

Time	Action	Message
1 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (1, 11, 16, 18, 2, 'Downtown Flat', '2.', 9, 1, 4, 173.91);	1 row(s) affected
2 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (2, 2, 1, 5, 5, 'Cozy Cottage', 'H.', 9, 1, 1, 188.63);	1 row(s) affected
3 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (3, 3, 10, 15, 15, 'Harbor Townhouse', 'N.', 6, 3, 4, 888.52);	1 row(s) affected
4 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (4, 10, 4, 3, 20, 'Hilltop Castle', 'Rick', 10, 7, 4, 651.67);	1 row(s) affected
5 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (5, 4, 12, 14, 20, 'Skyline Penthouse', 'Jared', 7, 5, 5, 192.08);	1 row(s) affected
6 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (6, 5, 3, 5, 16, 'Desert Oasis', 'H.', 10, 6, 2, 151.79);	1 row(s) affected
7 23:50:45	INSERT INTO property(id, address_id, host_id, place_type_id, property_type_id, property_name, property_description, max_num_guests, total_beds, total_bedrooms, price_per_night) VALUES (7, 11, 28, 18, 40977, '1-8 Long Lane', '3429 Farmview Lane');	1 row(s) affected

## booking

```

3  --
4  -- Inserting data into table 'airbnb','booking'
5  --
6  INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkout_date, booking_date, last_updated_at) VALUES
7  (1, 8, 2, 'SUCCESS', '2024-01-27', '2024-01-31', '2024-01-25', '2024-01-25 21:10:45.7956');
8  INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkout_date, booking_date, last_updated_at) VALUES
9  (2, 6, 3, 'SUCCESS', '2024-01-11', '2024-01-13', '2024-01-09', '2024-01-10 05:59:83.208662');
10 INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkout_date, booking_date, last_updated_at) VALUES
11 (3, 17, 10, 'SUCCESS', '2024-02-02', '2024-02-07', '2024-01-31', '2024-02-01 17:08:07.89114');
12 INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkin_date, booking_date, last_updated_at) VALUES
13 (4, 10, 11, 'SUCCESS', '2024-01-05', '2024-01-10', '2024-01-03', '2024-01-10 13:42:47.685613');
14 INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkout_date, booking_date, last_updated_at) VALUES
15 (5, 4, 15, 'SUCCESS', '2024-01-12', '2024-01-15', '2024-01-10', '2024-01-10 14:11:14.038014');
16 INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkout_date, booking_date, last_updated_at) VALUES
17 (6, 16, 22, 'SUCCESS', '2024-01-01', '2024-01-05', '2023-12-30', '2023-12-31 18:18:12.192714');
18 INSERT INTO booking(booking_id, property_id, user_id, booking_status, checkin_date, checkout_date, booking_date, last_updated_at) VALUES
19 (7, 0, 0, 'SUCCESS', '2024-01-01', '2024-01-01', '2024-01-01', '2024-01-01 00:00:00');

```

Time	Action	Message
1 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (1, 8, 2, 'SUCCESS',	1 row(s) affected
2 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (2, 6, 3, 'SUCCESS',	1 row(s) affected
3 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (3, 17, 10, 'SUCC	1 row(s) affected
4 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (4, 10, 11, 'SUCC	1 row(s) affected
5 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (5, 4, 15, 'SUCC	1 row(s) affected
6 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (6, 16, 22, 'SUCC	1 row(s) affected
7 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (7, 25, 'SUCC	1 row(s) affected
8 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (8, 7, 29, 'SUCC	1 row(s) affected
9 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (9, 14, 9, 'SUCC	1 row(s) affected
10 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (10, 6, 31, 'SUCC	1 row(s) affected
11 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (11, 2, 28, 'SUCC	1 row(s) affected
12 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (12, 12, 4, 'SUCC	1 row(s) affected
13 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (13, 17, 7, 'SUCC	1 row(s) affected
14 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (14, 9, 6, 'SUCC	1 row(s) affected
15 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (15, 11, 20, 'SUCC	1 row(s) affected
16 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (16, 17, 24, 'SUCC	1 row(s) affected
17 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (17, 14, 33, 'SUCC	1 row(s) affected
18 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (18, 6, 37, 'SUCC	1 row(s) affected
19 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (19, 5, 39, 'SUCC	1 row(s) affected
20 00:28:40	INSERT INTO booking(booking_id, property_id, user_id, booking_status, check_in_date, checkout_date, booking_date, last_updated_at) VALUES (20, 3, 4, 'SUCC	1 row(s) affected

## property\_review

```
--
-- Inserting data into table 'airbnb`.`property_review`
--
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(1, 18, 17, 'First and foremost, the results of the individual elements may share attitudes on every contradiction between the spec', '2024-03-15', 4, 3);
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(2, 11, 2, 'The efficiency of the change of marketing strategy gives less satisfactory results.', '2024-03-12', 3, 4);
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(3, 7, 3, 'Surprisingly, components of dimensions of the diverse sources of information discards the principle of the quality guide', '2024-03-15', 4, 3);
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(4, 1, 4, 'From these facts, one may conclude that the basic layout for the fundamental problem combines the feedback system and th', '2024-03-15', 4, 3);
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(5, 7, 5, 'It is undeniable that the total volume of the mechanism will require a vast knowledge.', '2024-03-15', 4, 3);
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(6, 7, 6, 'So far so good, but the application rules and growth opportunities of it are quite high.', '2024-03-18', 3, 4);
INSERT INTO property_review(review_id, property_id, booking_id, comment, review_date, host_rating, overall_rating) VALUES
(7, 2, 2, 2, 'The efficiency of the change of marketing strategy gives less satisfactory results.', '2024-03-12', 3, 4);
```

Time	Action	Message
1 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (1,18,17,'First and foremost, the res...	1 row(s) affected
2 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (2,11,2,'The the efficiency of the ch...	1 row(s) affected
3 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (3,7,3,'Surprisingly, components of ...	1 row(s) affected
4 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (4,1,4,'From these facts, one may c...	1 row(s) affected
5 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (5,4,7,5,'It is undeniable that the total...	1 row(s) affected
6 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (6,7,6,'So far so good, but the app...	1 row(s) affected
7 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (7,3,20,'To straighten it out, a pres...	1 row(s) affected
8 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (8,16,7,'Let's consider, that there is...	1 row(s) affected
9 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (9,7,16,'It is very clear from these ...	1 row(s) affected
10 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (10,7,8,'To put it simply, support t...	1 row(s) affected
11 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (11,9,1,'9 JUL 2024 03:08:45	1 row(s) affected
12 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (12,9,19,'Let's not forget that either...	1 row(s) affected
13 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (13,10,9,'At concerns details of the...	1 row(s) affected
14 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (14,14,10,'From these arguments on...	1 row(s) affected
15 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (15,15,5,'12, From these arguments one...	1 row(s) affected
16 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (16,3,14,'It is necessary to point out...	1 row(s) affected
17 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (17,4,15,'Let's consider, that eith...	1 row(s) affected
18 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (18,12,1,'It is worth emphasizing th...	1 row(s) affected
19 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (19,6,18,'Without a doubt, the t...	1 row(s) affected
20 00:29:22	INSERT INTO property_review(review_id,property_id,booking_id,comment,review_date,host_rating,overall_rating)VALUES (20,16,17,'As concerns elements of...	1 row(s) affected

# Data Insertion

## aspect\_rating

```
8 --
9 -- Inserting data into table 'airbnb`.`aspect_rating`
10 --
11
12 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
13   (1, 2, 2, 3);
14
15 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
16   (2, 4, 13, 1);
17
18 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
19   (3, 2, 13, 4);
20
21 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
22   (4, 1, 13, 5);
23
24 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
25   (5, 14, 12, 3);
26
27 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
28   (6, 15, 9, 3);
29
30 • INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES
31   (7, 11, 8, 5);
```

Action Output		
Time	Action	Message
1 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (1, 2, 2, 3)	1 row(s) affected
2 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (2, 4, 13, 1)	1 row(s) affected
3 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (3, 2, 13, 4)	1 row(s) affected
4 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (4, 1, 13, 5)	1 row(s) affected
5 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (5, 14, 12, 3)	1 row(s) affected
6 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (6, 15, 9, 3)	1 row(s) affected
7 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (7, 11, 8, 5)	1 row(s) affected
8 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (8, 2, 10, 3)	1 row(s) affected
9 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (9, 4, 3, 2)	1 row(s) affected
10 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (10, 10, 2, 5)	1 row(s) affected
11 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (11, 15, 18, 2)	1 row(s) affected
12 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (12, 9, 1, 2)	1 row(s) affected
13 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (13, 13, 4, 2)	1 row(s) affected
14 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (14, 11, 16, 3)	1 row(s) affected
15 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (15, 13, 1, 4)	1 row(s) affected
16 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (16, 14, 17, 4)	1 row(s) affected
17 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (17, 15, 10, 5)	1 row(s) affected
18 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (18, 19, 2, 3)	1 row(s) affected
19 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (19, 3, 18, 2)	1 row(s) affected
20 00:31:15	INSERT INTO aspect_rating(id, review_aspect_id, property_review_id, rating) VALUES (20, 16, 11, 3)	1 row(s) affected

## guest\_categories

```
7 --
8 -- Inserting data into table 'airbnb`.`guest_categories`
9 --
10
11 • INSERT INTO guest_categories(id, guest_type_name) VALUES
12   (1, 'Adults ');
13
14 • INSERT INTO guest_categories(id, guest_type_name) VALUES
15   (2, 'Children ');
16
17 • INSERT INTO guest_categories(id, guest_type_name) VALUES
18   (3, 'Infants ');
19
20 • INSERT INTO guest_categories(id, guest_type_name) VALUES
21   (4, 'Senior Citizens ');
22
23 • INSERT INTO guest_categories(id, guest_type_name) VALUES
24   (5, 'pets');
25
26 • INSERT INTO guest_categories(id, guest_type_name) VALUES
27   (6, 'Teens');
```

Action Output		
Time	Action	Message
1 00:32:16	INSERT INTO guest_categories(id, guest_type_name) VALUES (1, 'Adults ')	1 row(s) affected
2 00:32:16	INSERT INTO guest_categories(id, guest_type_name) VALUES (2, 'Children ')	1 row(s) affected
3 00:32:16	INSERT INTO guest_categories(id, guest_type_name) VALUES (3, 'Infants ')	1 row(s) affected
4 00:32:16	INSERT INTO guest_categories(id, guest_type_name) VALUES (4, 'Senior Citizens ')	1 row(s) affected
5 00:32:16	INSERT INTO guest_categories(id, guest_type_name) VALUES (5, 'pets')	1 row(s) affected
6 00:32:16	INSERT INTO guest_categories(id, guest_type_name) VALUES (6, 'Teens')	1 row(s) affected

# Data Insertion

## booking\_guest

```
--  
-- Inserting data into table 'airbnb'.booking_guest  
--  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (1, 12, 6, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (2, 1, 1, 7);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (3, 17, 2, 1);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (4, 12, 1, 7);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (5, 9, 1, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (6, 20, 1, 3);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (7, 15, 4, 5);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (8, 18, 1, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (9, 3, 4, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (10, 16, 1, 3);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (11, 8, 1, 3);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (12, 16, 6, 1);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (13, 13, 4, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (14, 10, 6, 3);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (15, 3, 1, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (16, 17, 1, 4);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (17, 13, 1, 4);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (18, 18, 6, 3);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (19, 12, 2, 2);  
• INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES  
  (20, 5, 6, 3);
```

Time	Action	Message
1 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (1, 12, 6, 2)	1 row(s) affected
2 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (2, 1, 1, 7)	1 row(s) affected
3 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (3, 17, 2, 1)	1 row(s) affected
4 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (4, 12, 1, 7)	1 row(s) affected
5 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (5, 9, 1, 2)	1 row(s) affected
6 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (6, 20, 1, 3)	1 row(s) affected
7 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (7, 15, 4, 5)	1 row(s) affected
8 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (8, 18, 1, 2)	1 row(s) affected
9 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (9, 3, 4, 2)	1 row(s) affected
10 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (10, 16, 1, 3)	1 row(s) affected
11 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (11, 8, 1, 3)	1 row(s) affected
12 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (12, 16, 6, 1)	1 row(s) affected
13 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (13, 13, 4, 2)	1 row(s) affected
14 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (14, 10, 6, 3)	1 row(s) affected
15 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (15, 3, 1, 2)	1 row(s) affected
16 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (16, 17, 1, 4)	1 row(s) affected
17 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (17, 13, 1, 4)	1 row(s) affected
18 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (18, 18, 6, 3)	1 row(s) affected
19 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (19, 12, 2, 2)	1 row(s) affected
20 00:33:08	INSERT INTO booking_guest(id, booking_id, guest_categories_id, num_guests) VALUES (20, 5, 6, 3)	1 row(s) affected

## attribute\_category

```
--  
-- Inserting data into table 'airbnb'.attribute_category  
--  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (1, 'Amenities', "This category captures the features and facilities offered within a property, such as Wi-Fi, a fully equipped kitchen, etc.");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (2, 'Pets', "This scope indicates whether pets are allowed in the property and outlines any related restrictions, like pet fees or breed restrictions.");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (3, 'View', "Surrounding scenery (beach, mountain, city)");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (4, 'Security', "Safety features (alarms, gated community)");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (5, 'Safety & Property', "Fire safety, first aid, maintenance");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (6, 'House Rules', "Guest regulations (smoking, quiet hours)");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (7, 'Services', "Additional offerings (concierge, housekeeping)");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (8, 'Accessibility', "Features for disabilities (ramps, wide doors)");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (9, 'Cancellation policy', "Booking cancellation terms");  
• INSERT INTO attribute_category(id, category_name, category_scope) VALUES  
  (10, 'Parking', "Parking options (on-site, street, fees)");
```

Time	Action	Message
1 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (1, 'Amenities', "This category captures the features and facilities offered within a property, such as Wi-Fi, a fully equipped kitchen, etc.");	1 row(s) affected
2 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (2, 'Pets', "This scope indicates whether pets are allowed in the property and outlines any related restrictions, like pet fees or breed restrictions.");	1 row(s) affected
3 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (3, 'View', "Surrounding scenery (beach, mountain, city)");	1 row(s) affected
4 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (4, 'Security', "Safety features (alarms, gated community)");	1 row(s) affected
5 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (5, 'Safety & Property', "Fire safety, first aid, maintenance");	1 row(s) affected
6 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (6, 'House Rules', "Guest regulations (smoking, quiet hours)");	1 row(s) affected
7 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (7, 'Services', "Additional offerings (concierge, housekeeping)");	1 row(s) affected
8 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (8, 'Accessibility', "Features for disabilities (ramps, wide doors)");	1 row(s) affected
9 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (9, 'Cancellation policy', "Booking cancellation terms");	1 row(s) affected
10 00:34:17	INSERT INTO attribute_category(id, category_name, category_scope) VALUES (10, 'Parking', "Parking options (on-site, street, fees)");	1 row(s) affected



# Data Insertion

## guest\_review

```
--
-- Inserting data into table `airbnb`.`guest_review`
--

INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(1, 11, 15, 'What is more, the internal resources minimizes influence of the proper evaluation of the application rules.', '2024-06-10', 4, 5);
INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(2, 12, 16, 'Furthermore, one should not forget that there is a direct relation between the base configuration and a broad understanding of the system.', '2024-06-11', 5, 5);
INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(3, 4, 5, 'Fortunately, violations of the big impact the major outcomes.', '2024-06-30', 2, 5);
INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(4, 3, 10, 'Moreover, the core principles must be compatible with the strategic decisions.', '2024-06-14', 1, 2);
INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(5, 10, 20, 'For a number of reasons, the valuable information becomes extremely important for the final draft.', '2024-06-12', 4, 5);
INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(6, 12, 19, 'Notwithstanding that the initial progress in the individual elements cannot rely only on the data management and data analysis, the system is still in the early stages of development and requires further refinement and testing.');
INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES
(7, 17, 14, 'On the contrary, within the framework of the current system, the data management and data analysis are still in the early stages of development and require further refinement and testing.');
```

Time	Action	Message
1 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (1, 11, 15, 'What is more, the internal resources minimizes influence of the proper evaluation of the application rules.', '2024-06-10', 4, 5);	1 row(s) affected
2 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (2, 12, 16, 'Furthermore, one should not forget that there is a direct relation between the base configuration and a broad understanding of the system.', '2024-06-11', 5, 5);	1 row(s) affected
3 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (3, 4, 5, 'Fortunately, violations of the big impact the major outcomes.', '2024-06-30', 2, 5);	1 row(s) affected
4 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (4, 3, 10, 'Moreover, the core principles must be compatible with the strategic decisions.', '2024-06-14', 1, 2);	1 row(s) affected
5 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (5, 10, 20, 'For a number of reasons, the valuable information becomes extremely important for the final draft.', '2024-06-12', 4, 5);	1 row(s) affected
6 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (6, 12, 19, 'Notwithstanding that the initial progress in the individual elements cannot rely only on the data management and data analysis, the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
7 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (7, 17, 14, 'On the contrary, within the framework of the current system, the data management and data analysis are still in the early stages of development and require further refinement and testing.');	1 row(s) affected
8 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (8, 2, 4, 'On top of that with the exception of the fact that the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
9 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (9, 3, 2, 'Besides, segments of the essence provided by the system are still in the early stages of development and require further refinement and testing.');	1 row(s) affected
10 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (10, 17, 7, 'Frankly speaking, the edge of the essence provided by the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
11 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (11, 3, 10, 'To all effects and purposes, the problem of the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
12 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (12, 11, 11, 'NULL, 2024-06-19', 3, 4);	1 row(s) affected
13 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (13, 10, 20, 'Surprisingly, in terms of the essential characteristics of the system, the data management and data analysis are still in the early stages of development and require further refinement and testing.');	1 row(s) affected
14 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (14, 18, 1, 'The most common argument against the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
15 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (15, 12, 16, 'One cannot possibly accept the fact that the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
16 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (16, 12, 16, 'Quite possibly, with the exception of the fact that the system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
17 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (17, 16, 8, 'One of the most striking features of this system is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected
18 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (18, 7, 12, 'In the meantime all approaches to the system are still in the early stages of development and require further refinement and testing.');	1 row(s) affected
19 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (19, 4, 5, 'To straighten it out, the core principles themselves are still in the early stages of development and require further refinement and testing.');	1 row(s) affected
20 00:35:58	INSERT INTO guest_review(id, host_id, booking_id, comment, review_date, guest_rating, overall_rating) VALUES (20, 17, 14, 'Alas, the evaluation of reliability activities is still in the early stages of development and requires further refinement and testing.');	1 row(s) affected

## favourite

```
--
-- Inserting data into table `airbnb`.`favourite`
--

INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(1, 10, 18, '2023-12-22 12:05:17');
INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(2, 10, 21, '2023-12-27 10:21:42');
INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(3, 11, 16, '2023-12-20 15:49:33');
INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(4, 8, 31, '2023-12-28 04:37:59');
INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(5, 15, 37, '2023-12-29 18:53:24');
INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(6, 15, 16, '2023-12-21 07:14:08');
INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES
(7, 17, 14, '2023-12-20 00:10:04');
```

Time	Action	Message
1 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (1, 10, 18, '2023-12-22 12:05:17');	1 row(s) affected
2 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (2, 10, 21, '2023-12-27 10:21:42');	1 row(s) affected
3 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (3, 11, 16, '2023-12-20 15:49:33');	1 row(s) affected
4 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (4, 8, 31, '2023-12-28 04:37:59');	1 row(s) affected
5 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (5, 15, 37, '2023-12-29 18:53:24');	1 row(s) affected
6 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (6, 15, 16, '2023-12-21 07:14:08');	1 row(s) affected
7 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (7, 17, 14, '2023-12-20 00:10:04');	1 row(s) affected
8 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (8, 15, 10, '2023-12-20 00:10:04');	1 row(s) affected
9 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (9, 13, 12, '2023-12-26 08:28:56');	1 row(s) affected
10 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (10, 14, 32, '2023-12-30 23:49:21');	1 row(s) affected
11 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (11, 30, 2023-12-29 09:15:47);	1 row(s) affected
12 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (12, 16, 6, '2023-12-22 02:31:45');	1 row(s) affected
13 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (13, 13, 26, '2024-01-01 04:12:13');	1 row(s) affected
14 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (14, 11, 15, '2023-12-28 15:04:30');	1 row(s) affected
15 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (15, 1, 7, '2023-12-19 21:40:20');	1 row(s) affected
16 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (16, 1, 33, '2023-12-31 05:05:58');	1 row(s) affected
17 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (17, 1, 13, '2023-12-23 14:19:36');	1 row(s) affected
18 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (18, 2, 28, '2023-12-24 07:50:01');	1 row(s) affected
19 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (19, 10, 18, '2023-12-27 02:59:14');	1 row(s) affected
20 00:35:20	INSERT INTO favourite(id, property_id, user_id, date_favorited) VALUES (20, 12, 34, '2023-12-20 08:35:39');	1 row(s) affected

# Data Insertion

## message

```
9 --
10 -- Inserting data into table 'airbnb`.`messages`
11 --
12 1 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
13 (1, 34, 5, 'It is obvious, that dimensions of the internal policy the global management concepts.', '2024-01-01 00:07:03');
14 2 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
15 (2, 13, 36, 'Conversely, a surprising flexibility in segments of the standards control provides rich insights into an initial attempt
16 3 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
17 (3, 1, 14, 'To put it simply, a huge improvement of the skills has more common features with every contradiction between the perform
18 4 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
19 (4, 27, 38, 'It is necessary to point out that the influence of the relation between the development process and the grand strategy
20 5 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
21 (5, 3, 21, 'As a matter of fact, the remainder of the set of system properties can be regarded as rigorously insignificant.', '2024-
22 6 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
23 (6, 8, 10, 'Let's not forget that the negative impact of the crucial component cannot be developed under such circumstances.', '202
24 7 INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES
25 (7, 4, 23, 'In a similar manner, the unification of the tasks priority man...
```

Time	Action	Message
1 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (1, 34, 5, 'It is obvious, that dimensions of the internal policy the glob...	1 row(s) affected
2 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (2, 13, 36, 'Conversely, a surprising flexibility in segments of the stan...	1 row(s) affected
3 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (3, 1, 14, 'To put it simply, a huge improvement of the skills has more...	1 row(s) affected
4 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (4, 27, 38, 'It is necessary to point out that the influence of the rela...	1 row(s) affected
5 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (5, 3, 21, 'As a matter of fact, the remainder of the set of system pro...	1 row(s) affected
6 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (6, 8, 10, 'Let's not forget that the negative impact of the crucial co...	1 row(s) affected
7 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (7, 4, 23, 'In a similar manner, the unification of the tasks priority man...	1 row(s) affected
8 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (8, 38, 9, 'It is worth emphasizing that the design of the comprehensi...	1 row(s) affected
9 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (9, 25, 38, 'According to some experts, the advantage of the basic f...	1 row(s) affected
10 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (10, 2, 17, 'As concerns a broad understanding of the structured t...	1 row(s) affected
11 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (11, 14, 3, 'To all effects and purposes, the organization of the inter...	1 row(s) affected
12 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (12, 8, 27, 'Notwithstanding that the efficiency of the internal policy L...	1 row(s) affected
13 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (13, 39, 36, 'That is to say the point of the essence benefits from per...	1 row(s) affected
14 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (14, 38, 35, 'Otherwise speaking, impact of the treatment enforces t...	1 row(s) affected
15 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (15, 29, 6, 'In respect that any further consideration is regularly deba...	1 row(s) affected
16 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (16, 32, 16, 'As a resultant implication, elements of the mechanism s...	1 row(s) affected
17 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (17, 28, 36, 'It is obvious, that the core principles has more common...	1 row(s) affected
18 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (18, 30, 32, 'One should, nevertheless, consider that a growth of th...	1 row(s) affected
19 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (19, 14, 23, 'In a word, the application interface and session operatio...	1 row(s) affected
20 00:40:35	INSERT INTO messages(id, sender_id, receiver_id, message_content, sent_timestamp) VALUES (20, 14, 21, 'It should not be neglected that a closer study of the be...	1 row(s) affected

## language

```
9 --
10 -- Inserting data into table 'airbnb`.`language`
11 --
12 1 INSERT INTO language(id, language_name) VALUES
13 (1, 'Vietnamese ');
14 2 INSERT INTO language(id, language_name) VALUES
15 (2, 'Bengali ');
16 3 INSERT INTO language(id, language_name) VALUES
17 (3, 'Urdu ');
18 4 INSERT INTO language(id, language_name) VALUES
19 (4, 'Finnish ');
20 5 INSERT INTO language(id, language_name) VALUES
21 (5, 'Mandarin Chinese ');
22 6 INSERT INTO language(id, language_name) VALUES
23 (6, 'Serbian ');
24 7 INSERT INTO language(id, language_name) VALUES
25 (7, 'Japanese ');
26 8 INSERT INTO language(id, language_name) VALUES
27 (8, 'Arabic standard ');
28 9 INSERT INTO language(id, language_name) VALUES
29 (9, 'Dutch ');
30 10 INSERT INTO language(id, language_name) VALUES
31 (10, 'English ');
32 11 INSERT INTO language(id, language_name) VALUES
33 (11, 'Portuguese ');
34 12 INSERT INTO language(id, language_name) VALUES
35 (12, 'Croatian ');
36 13 INSERT INTO language(id, language_name) VALUES
37 (13, 'German ');
38 14 INSERT INTO language(id, language_name) VALUES
39 (14, 'Norwegian ');
40 15 INSERT INTO language(id, language_name) VALUES
41 (15, 'Marathi ');
42 16 INSERT INTO language(id, language_name) VALUES
43 (16, 'Telugu ');
44 17 INSERT INTO language(id, language_name) VALUES
45 (17, 'Polish ');
46 18 INSERT INTO language(id, language_name) VALUES
47 (18, 'Hebrew ');
48 19 INSERT INTO language(id, language_name) VALUES
49 (19, 'Hindi ');
50 20 INSERT INTO language(id, language_name) VALUES
51 (20, 'Persian ');
```

Time	Action	Message
1 00:36:58	INSERT INTO language(id, language_name) VALUES (1, 'Vietnamese ')	1 row(s) affected
2 00:36:58	INSERT INTO language(id, language_name) VALUES (2, 'Bengali ')	1 row(s) affected
3 00:36:58	INSERT INTO language(id, language_name) VALUES (3, 'Urdu ')	1 row(s) affected
4 00:36:58	INSERT INTO language(id, language_name) VALUES (4, 'Finnish ')	1 row(s) affected
5 00:36:58	INSERT INTO language(id, language_name) VALUES (5, 'Mandarin Chinese ')	1 row(s) affected
6 00:36:58	INSERT INTO language(id, language_name) VALUES (6, 'Serbian ')	1 row(s) affected
7 00:36:58	INSERT INTO language(id, language_name) VALUES (7, 'Japanese ')	1 row(s) affected
8 00:36:58	INSERT INTO language(id, language_name) VALUES (8, 'Arabic standard ')	1 row(s) affected
9 00:36:58	INSERT INTO language(id, language_name) VALUES (9, 'Dutch ')	1 row(s) affected
10 00:36:58	INSERT INTO language(id, language_name) VALUES (10, 'English ')	1 row(s) affected
11 00:36:58	INSERT INTO language(id, language_name) VALUES (11, 'Portuguese ')	1 row(s) affected
12 00:36:58	INSERT INTO language(id, language_name) VALUES (12, 'Croatian ')	1 row(s) affected
13 00:36:58	INSERT INTO language(id, language_name) VALUES (13, 'German ')	1 row(s) affected
14 00:36:58	INSERT INTO language(id, language_name) VALUES (14, 'Norwegian ')	1 row(s) affected
15 00:36:58	INSERT INTO language(id, language_name) VALUES (15, 'Marathi ')	1 row(s) affected
16 00:36:58	INSERT INTO language(id, language_name) VALUES (16, 'Telugu ')	1 row(s) affected
17 00:36:58	INSERT INTO language(id, language_name) VALUES (17, 'Polish ')	1 row(s) affected
18 00:36:58	INSERT INTO language(id, language_name) VALUES (18, 'Hebrew ')	1 row(s) affected
19 00:36:58	INSERT INTO language(id, language_name) VALUES (19, 'Hindi ')	1 row(s) affected
20 00:36:58	INSERT INTO language(id, language_name) VALUES (20, 'Persian ')	1 row(s) affected

# Data Insertion

## property\_availability

```
9 --
10 -- Inserting data into table `airbnb`.`property_availability`
11 --
12
13 • INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES
14 (1, 10, '2024-05-01', '2024-05-14');
15
16 • INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES
17 (2, 3, '2024-05-01', '2024-05-14');
18
19 • INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES
20 (3, 3, '2024-05-02', '2024-05-11');
21
22 • INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES
23 (4, 1, '2024-05-01', '2024-05-13');
24
25 • INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES
26 (5, 16, '2024-05-03', '2024-05-12');
27
28 • INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES
29 (6, 1, '2024-05-05', '2024-05-15');
```

Action Output			
Time	Action	Message	
1 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (1, 10, '2024-05-01', '2024-05-14')	1 row(s) affected	
2 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (2, 3, '2024-05-01', '2024-05-14')	1 row(s) affected	
3 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (3, 3, '2024-05-02', '2024-05-11')	1 row(s) affected	
4 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (4, 1, '2024-05-01', '2024-05-13')	1 row(s) affected	
5 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (5, 16, '2024-05-03', '2024-05-12')	1 row(s) affected	
6 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (6, 1, '2024-05-05', '2024-05-15')	1 row(s) affected	
7 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (7, 17, '2024-05-01', '2024-05-13')	1 row(s) affected	
8 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (8, 20, '2024-05-01', '2024-05-13')	1 row(s) affected	
9 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (9, 8, '2024-05-04', '2024-05-11')	1 row(s) affected	
10 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (10, 7, '2024-05-05', '2024-05-11')	1 row(s) affected	
11 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (11, 7, '2024-05-02', '2024-05-15')	1 row(s) affected	
12 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (12, 4, '2024-05-06', '2024-05-12')	1 row(s) affected	
13 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (13, 3, '2024-05-03', '2024-05-14')	1 row(s) affected	
14 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (14, 1, '2024-05-05', '2024-05-14')	1 row(s) affected	
15 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (15, 15, '2024-05-05', '2024-05-15')	1 row(s) affected	
16 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (16, 16, '2024-05-05', '2024-05-12')	1 row(s) affected	
17 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (17, 7, '2024-05-08', '2024-05-15')	1 row(s) affected	
18 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (18, 4, '2024-05-06', '2024-05-14')	1 row(s) affected	
19 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (19, 6, '2024-05-01', '2024-05-15')	1 row(s) affected	
20 00:41:09	INSERT INTO property_availability(id, property_id, start_date, end_date) VALUES (20, 16, '2024-05-05', '2024-05-13')	1 row(s) affected	

## property\_category

```
9 --
10 -- Inserting data into table `airbnb`.`property_category`
11 --
12
13 • INSERT INTO property_category(id, category_name) VALUES
14 (1, 'surfing');
15
16 • INSERT INTO property_category(id, category_name) VALUES
17 (2, 'top of the hill');
18
19 • INSERT INTO property_category(id, category_name) VALUES
20 (3, 'amazing view');
21
22 • INSERT INTO property_category(id, category_name) VALUES
23 (4, 'camp');
24
25 • INSERT INTO property_category(id, category_name) VALUES
26 (5, 'Top of the world');
27
28 • INSERT INTO property_category(id, category_name) VALUES
29 (6, 'skiing');
```

Action Output		
Time	Action	Message
1 00:41:56	INSERT INTO property_category(id, category_name) VALUES (1, 'surfing')	1 row(s) affected
2 00:41:56	INSERT INTO property_category(id, category_name) VALUES (2, 'top of the hill')	1 row(s) affected
3 00:41:56	INSERT INTO property_category(id, category_name) VALUES (3, 'amazing view')	1 row(s) affected
4 00:41:56	INSERT INTO property_category(id, category_name) VALUES (4, 'camp')	1 row(s) affected
5 00:41:56	INSERT INTO property_category(id, category_name) VALUES (5, 'Top of the world')	1 row(s) affected
6 00:41:56	INSERT INTO property_category(id, category_name) VALUES (6, 'skiing')	1 row(s) affected
7 00:41:56	INSERT INTO property_category(id, category_name) VALUES (7, 'national park')	1 row(s) affected
8 00:41:56	INSERT INTO property_category(id, category_name) VALUES (8, 'arctic')	1 row(s) affected
9 00:41:56	INSERT INTO property_category(id, category_name) VALUES (9, 'mansion')	1 row(s) affected
10 00:41:56	INSERT INTO property_category(id, category_name) VALUES (10, 'country side')	1 row(s) affected
11 00:41:56	INSERT INTO property_category(id, category_name) VALUES (11, 'Lake')	1 row(s) affected
12 00:41:56	INSERT INTO property_category(id, category_name) VALUES (12, 'creative')	1 row(s) affected
13 00:41:56	INSERT INTO property_category(id, category_name) VALUES (13, 'island')	1 row(s) affected
14 00:41:56	INSERT INTO property_category(id, category_name) VALUES (14, 'omg!')	1 row(s) affected
15 00:41:56	INSERT INTO property_category(id, category_name) VALUES (15, 'iconic cities')	1 row(s) affected
16 00:41:56	INSERT INTO property_category(id, category_name) VALUES (16, 'tropical')	1 row(s) affected
17 00:41:56	INSERT INTO property_category(id, category_name) VALUES (17, 'earth homes')	1 row(s) affected
18 00:41:56	INSERT INTO property_category(id, category_name) VALUES (18, 'beach')	1 row(s) affected
19 00:41:56	INSERT INTO property_category(id, category_name) VALUES (19, 'trending')	1 row(s) affected
20 00:41:56	INSERT INTO property_category(id, category_name) VALUES (20, 'cabin')	1 row(s) affected

# Data Insertion

## property\_category\_link

```
--
-- Inserting data into table 'airbnb`.`property_category_link`
--
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(1, 13, 12);
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(2, 8, 15);
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(3, 11, 20);
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(4, 10, 5);
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(5, 11, 1);
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(6, 14, 2);
INSERT INTO property_category_link(id, property_id, property_category_id) VALUES
(7, 14, 2);
```

Time	Action	Message
1 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (1, 13, 12)	1 row(s) affected
2 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (2, 8, 15)	1 row(s) affected
3 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (3, 11, 20)	1 row(s) affected
4 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (4, 10, 5)	1 row(s) affected
5 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (5, 11, 1)	1 row(s) affected
6 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (6, 14, 2)	1 row(s) affected
7 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (7, 3, 10)	1 row(s) affected
8 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (8, 5, 3)	1 row(s) affected
9 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (9, 3, 12)	1 row(s) affected
10 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (10, 7, 17)	1 row(s) affected
11 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (11, 3, 4)	1 row(s) affected
12 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (12, 8, 3)	1 row(s) affected
13 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (13, 6, 7)	1 row(s) affected
14 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (14, 10, 18)	1 row(s) affected
15 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (15, 4, 9)	1 row(s) affected
16 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (16, 4, 8)	1 row(s) affected
17 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (17, 11, 15)	1 row(s) affected
18 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (18, 6, 4)	1 row(s) affected
19 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (19, 15, 11)	1 row(s) affected
20 00:43:18	INSERT INTO property_category_link(id, property_id, property_category_id) VALUES (20, 20, 3)	1 row(s) affected

## property\_attributes

```
--
-- Inserting data into table 'airbnb`.`property_attributes`
--
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(1, 9, 'Flexible', 'Free cancellation up to 24 hours before check-in');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(2, 9, 'Moderate', 'Free cancellation up to 5 days before check-in');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(3, 9, 'Strict', 'No free cancellation; full refund only if canceled within 48 hours of booking');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(4, 2, 'Pets Allowed', 'Guests can bring their pets');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(5, 2, 'No Pets', 'Pets are not allowed');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(6, 3, 'Ocean View', 'Enjoy a stunning view of the ocean');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(7, 3, 'Mountain View', 'Enjoy a stunning view of the surrounding mountains');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(8, 4, 'Security Cameras', '24/7 surveillance for added security');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(9, 4, 'Gated Community', 'Secure access with gated entry');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(10, 5, 'Smoke Detectors', 'Installed for early fire detection');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(11, 5, 'Fire Extinguishers', 'Available for emergencies');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(12, 6, 'No Smoking', 'Strict no-smoking policy inside the property');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(13, 6, 'Quiet Hours', 'Observed for peaceful stay');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(14, 7, 'Housekeeping', 'Regular cleaning and tidying services available');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(15, 7, 'Airport Transfers', 'Convenient transportation to/from the airport');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(16, 8, 'Ramp', 'Ramp available for wheelchair access');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(17, 8, 'Accessible Bathroom', 'Accessible bathroom with grab bars and shower chair');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(18, 6, 'Quiet Neighborhood', 'Peaceful surroundings away from busy streets');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(19, 1, 'Swimming Pool', 'Private pool for guests');
INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES
(20, 1, 'Tennis Court', 'Outdoor tennis court for sports enthusiasts');
```

Time	Action	Message
1 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (1, 9, 'Flexible', 'Free cancellation up to 24 hours before check-in')	1 row(s) affected
2 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (2, 9, 'Moderate', 'Free cancellation up to 5 days before check-in')	1 row(s) affected
3 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (3, 9, 'Strict', 'No free cancellation; full refund only if canceled within 48 hours of booking')	1 row(s) affected
4 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (4, 2, 'Pets Allowed', 'Guests can bring their pets')	1 row(s) affected
5 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (5, 2, 'No Pets', 'Pets are not allowed')	1 row(s) affected
6 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (6, 3, 'Ocean View', 'Enjoy a stunning view of the ocean')	1 row(s) affected
7 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (7, 3, 'Mountain View', 'Enjoy a stunning view of the surrounding mountains')	1 row(s) affected
8 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (8, 4, 'Security Cameras', '24/7 surveillance for added security')	1 row(s) affected
9 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (9, 4, 'Gated Community', 'Secure access with gated entry')	1 row(s) affected
10 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (10, 5, 'Smoke Detectors', 'Installed for early fire detection')	1 row(s) affected
11 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (11, 5, 'Fire Extinguishers', 'Available for emergencies')	1 row(s) affected
12 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (12, 6, 'No Smoking', 'Strict no-smoking policy inside the property')	1 row(s) affected
13 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (13, 6, 'Quiet Hours', 'Observed for peaceful stay')	1 row(s) affected
14 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (14, 7, 'Housekeeping', 'Regular cleaning and tidying services available')	1 row(s) affected
15 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (15, 7, 'Airport Transfers', 'Convenient transportation to/from the airport')	1 row(s) affected
16 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (16, 8, 'Ramp', 'Ramp available for wheelchair access')	1 row(s) affected
17 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (17, 8, 'Accessible Bathroom', 'Accessible bathroom with grab bars and shower chair')	1 row(s) affected
18 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (18, 6, 'Quiet Neighborhood', 'Peaceful surroundings away from busy streets')	1 row(s) affected
19 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (19, 1, 'Swimming Pool', 'Private pool for guests')	1 row(s) affected
20 00:46:42	INSERT INTO property_attributes(id, attribute_category_id, attribute_name, attribute_description) VALUES (20, 1, 'Tennis Court', 'Outdoor tennis court for sports enthusiasts')	1 row(s) affected

# Data Insertion

## social\_media

```
--
-- Inserting data into table 'airbnb`.`social_media`

INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
(1, 30, 'Facebook', 'http://www.butedoul.fr/edfote/anthi/meas/henenomee.php?t=69');
INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
(2, 24, 'Instagram', 'https://tieandsho.se/as/th/thiwasteen.aspx?id=498');
INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
(3, 11, 'Instagram', 'http://eraaromees.br/andes/butithti/ing/enan.php?t=55&p=81');
INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
(4, 30, 'Youtube', 'https://youtube.com/areatall/ngevehad/ourareted/steatoes.asp');
INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
(5, 12, 'Instagram', 'https://eveherhaent.es/metise/ourbutndse.htm');
INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
(6, 35, 'Youtube', 'https://youtube.com/henthaster323/1231jaF234');
INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES
```

Time	Action	Message
1 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (1, 30, 'Facebook', 'http://www.butedoul.fr/edfote/anthi/meas/henenomee...)	1 row(s) affected
2 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (2, 24, 'Instagram', 'https://tieandsho.se/as/th/thiwasteen.aspx?id=498')	1 row(s) affected
3 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (3, 11, 'Instagram', 'http://eraaromees.br/andes/butithti/ing/enan.php?t=55&p=81')	1 row(s) affected
4 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (4, 30, 'Youtube', 'https://youtube.com/areatall/ngevehad/ourareted/steatoes.asp')	1 row(s) affected
5 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (5, 12, 'Instagram', 'https://eveherhaent.es/metise/ourbutndse.htm')	1 row(s) affected
6 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (6, 35, 'Youtube', 'https://youtube.com/henthaster323/1231jaF234')	1 row(s) affected
7 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (7, 30, 'LinkedIn', 'https://www.linkedin.com/company/areatall/ngevehad/ourareted/steatoes.asp')	1 row(s) affected
8 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (8, 9, 'Facebook', 'https://www.facebook.com/henthaster323/1231jaF234')	1 row(s) affected
9 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (9, 33, 'Instagram', 'https://www.instagram.com/henthaster323/1231jaF234')	1 row(s) affected
10 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (10, 21, 'LinkedIn', 'https://www.linkedin.com/company/henthaster323/1231jaF234')	1 row(s) affected
11 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (11, 37, 'Youtube', 'https://youtube.com/32and34et/hyio/taoual')	1 row(s) affected
12 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (12, 28, 'Instagram', 'https://www.wali.g/thahadr/sho/eth/iei.php?t=69')	1 row(s) affected
13 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (13, 1, 'Youtube', 'https://youtube.com/niseethi.php?t=22&p=8968')	1 row(s) affected
14 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (14, 31, 'LinkedIn', 'https://www.linkedin.com/company/niseethi.php?t=22&p=8968')	1 row(s) affected
15 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (15, 25, 'Facebook', 'https://www.facebook.com/niseethi.php?t=22&p=8968')	1 row(s) affected
16 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (16, 7, 'Instagram', 'https://www.instagram.com/niseethi.php?t=22&p=8968')	1 row(s) affected
17 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (17, 10, 'Youtube', 'https://youtube.com/niseethi.php?t=22&p=8968')	1 row(s) affected
18 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (18, 6, 'Instagram', 'https://www.instagram.com/niseethi.php?t=22&p=8968')	1 row(s) affected
19 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (19, 26, 'LinkedIn', NULL)	1 row(s) affected
20 00:48:35	INSERT INTO social_media(id, user_account_id, platform_name, account_url) VALUES (20, 5, 'Instagram', 'https://www.instagram.com/niseethi.php?t=22&p=8968')	1 row(s) affected

## property\_amenities\_rules

```
--
-- Inserting data into table 'airbnb`.`property_amenities_rules`

INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
(1, 15, 1, "2024-01-12");
INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
(2, 9, 2, "2023-12-20");
INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
(3, 2, 3, "2024-01-27");
INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
(4, 1, 23, "2024-02-13");
INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
(5, 4, 21, "2024-01-23");
INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
(6, 14, 15, "2024-02-07");
INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES
```

Time	Action	Message
1 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (1, 15, 1, "2024-01-12")	1 row(s) affected
2 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (2, 9, 2, "2023-12-20")	1 row(s) affected
3 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (3, 2, 3, "2024-01-27")	1 row(s) affected
4 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (4, 1, 23, "2024-02-13")	1 row(s) affected
5 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (5, 4, 21, "2024-01-23")	1 row(s) affected
6 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (6, 14, 15, "2024-02-07")	1 row(s) affected
7 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (7, 12, 14, "2024-01-04")	1 row(s) affected
8 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (8, 17, 11, "2023-12-28")	1 row(s) affected
9 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (9, 7, 10, "2024-02-19")	1 row(s) affected
10 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (10, 10, 9, "2024-01-08")	1 row(s) affected
11 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (11, 13, 1, "2023-12-14")	1 row(s) affected
12 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (12, 15, 3, "2024-02-21")	1 row(s) affected
13 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (13, 2, 9, "2024-01-10")	1 row(s) affected
14 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (14, 14, 23, "2023-12-08")	1 row(s) affected
15 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (15, 10, 30, "2024-02-05")	1 row(s) affected
16 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (16, 13, 31, "2024-01-18")	1 row(s) affected
17 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (17, 10, 2, "2023-12-19")	1 row(s) affected
18 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (18, 2, 14, "2023-12-05")	1 row(s) affected
19 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (19, 5, 16, "2024-02-15")	1 row(s) affected
20 00:47:51	INSERT INTO property_amenities_rules(id, property_id, property_attributes_id, last_updated) VALUES (20, 5, 17, "2024-01-25")	1 row(s) affected

# voucher

Time	Action	Message
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (1,'Extended Stay Special','3P0W...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (2,'Extended Stay Escape','7BC1...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (3,'Adventure Seeker's Paradise','...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (4,'Nature's Embrace','4EYD0H...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (5,'Book Direct Bliss','3P20DXD...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (6,'Trekking offer','vMgKRP04','...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (7,'Adventure Seeker's Paradise','...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (8,'Memories in the Making','TQJ8...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (9,'Foodie Adventure Special','vR...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (10,'Extended Stay Awards','DM1...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (11,'Romantic Escape for Two','4...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (12,'Tranquility Retreat','8L0Q0P3...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (13,'Cozy Cabin Getaways','NTST1...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (14,'Family Fun Fiesta','EMPFEG...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (15,'Group Getaway Getaways','1U...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (16,','QBBCY20','81.91.1854...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (17,'Group Getaway Getaways','FJ...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (18,'Midweek Recharge','SHZCA...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (19,'Book Direct Bliss','56LRCGCV...	1 row(s) affected
00:49:57	INSERT INTO voucher(voucher_id,title,unique_code,discount_amount,min_booking_value,expiry_date,created_at)VALUES (20,'Group Getaway Getaways','47...	1 row(s) affected

## user\_language

Time	Action	Message
1 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (1, 27, 10)	1 row(s) affected
2 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (2, 3, 10)	1 row(s) affected
3 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (3, 11, 31)	1 row(s) affected
4 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (4, 16, 6)	1 row(s) affected
5 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (5, 27, 1)	1 row(s) affected
6 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (6, 21, 20)	1 row(s) affected
7 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (7, 33, 17)	1 row(s) affected
8 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (8, 12, 10)	1 row(s) affected
9 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (9, 39, 10)	1 row(s) affected
10 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (10, 36, 1)	1 row(s) affected
11 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (11, 13, 10)	1 row(s) affected
12 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (12, 37, 29)	1 row(s) affected
13 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (13, 7, 10)	1 row(s) affected
14 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (14, 1, 25)	1 row(s) affected
15 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (15, 34, 10)	1 row(s) affected
16 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (16, 22, 10)	1 row(s) affected
17 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (17, 25, 31)	1 row(s) affected
18 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (18, 2, 15)	1 row(s) affected
19 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (19, 21, 10)	1 row(s) affected
20 00:53:27	INSERT INTO user_language(id, user_id, language_id) VALUES (20, 37, 19)	1 row(s) affected

```
--
-- Inserting data into table `airbnb`.`transaction`
--
1 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
2 (1, 10, NULL, 'PhonePe', '2024-02-10', 10.16, 25, 11.46, 26.36, 'SUCCESS');
3
4 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
5 (2, 20, 6, 'PhonePe', '2024-02-11', 11.6, 21.85, 12.41, 21.42, 'SUCCESS');
6
7 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
8 (3, 17, 4, 'Direct Debit', '2024-02-11', 11.6, 20.04, 14.1, 20.18, 'SUCCESS');
9
10 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
11 (4, 3, 9, 'Direct Debit', '2024-02-11', 11.05, 22.42, 10.77, 15.83, 'SUCCESS');
12
13 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
14 (5, 1, NULL, 'Credit Card', '2024-02-11', 13, 24.55, 13.9, 34.65, 'SUCCESS');
15
16 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
17 (6, 13, NULL, 'WebMoney', '2024-02-10', 10.28, 22.49, 14.7, 25.79, 'SUCCESS');
18
19 INSERT INTO transaction(id, booking_id, voucher_id, payment_method, transaction_date, tax, service_charge, cleaning_charge, airbnb_charge, transaction_status) VALUES
20 (7, 4, NULL, 'PhonePe', '2024-02-10', 10.16, 25, 11.46, 26.36, 'SUCCESS');
```

[illegible]



# Data Insertion

## user\_preferred\_guest\_type

```
3  --
4  -- Inserting data into table 'airbnb'.user_preferred_guest_type
5  --
6  * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (1, 2, 1, 2);
7  * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (2, 1, 2, 2);
8  * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (3, 4, 3, 2);
9  * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (4, 3, 4, 1);
10 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (5, 2, 5, 3);
11 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (6, 1, 6, 4);
12 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (7, 3, 1, 1);
13 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (8, 2, 6, 3);
14 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (9, 5, 6, 0);
15 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (10, 6, 1, 2);
16 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (11, 7, 1, 3);
17 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (12, 21, 1, 2);
18 * INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (13, 22, 1, 3);
19 --
```

Time	Action	Message
1 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (1, 2, 1, 2)	1 row(s) affected
2 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (2, 1, 2, 2)	1 row(s) affected
3 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (3, 4, 3, 2)	1 row(s) affected
4 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (4, 3, 4, 1)	1 row(s) affected
5 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (5, 2, 5, 3)	1 row(s) affected
6 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (6, 1, 6, 4)	1 row(s) affected
7 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (7, 3, 1, 1)	1 row(s) affected
8 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (8, 2, 6, 3)	1 row(s) affected
9 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (9, 5, 6, 0)	1 row(s) affected
10 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (10, 6, 1, 2)	1 row(s) affected
11 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (11, 7, 1, 3)	1 row(s) affected
12 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (12, 21, 1, 2)	1 row(s) affected
13 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (13, 22, 1, 3)	1 row(s) affected
14 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (14, 26, 4, 2)	1 row(s) affected
15 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (15, 38, 1, 2)	1 row(s) affected
16 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (16, 40, 1, 3)	1 row(s) affected
17 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (17, 17, 1, 2)	1 row(s) affected
18 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (18, 35, 5, 2)	1 row(s) affected
19 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (19, 37, 1, 2)	1 row(s) affected
20 00:54:08	INSERT INTO user_preferred_guest_type (id, user_account_id, guest_categories_id, preferred_num_guests) VALUES (20, 34, 1, 2)	1 row(s) affected

## neighborhood

```
0  --
1  -- Inserting data into table 'airbnb'.neighborhood
2  --
3  * INSERT INTO neighborhood (id, location_id, neighborhood_name, neighborhood_description, transportation_score, safety_rating) VALUES
4  (1, 1, 'Colaba', 'A vibrant neighborhood in Mumbai known for its historic buildings and bustling markets.', 8, 7),
5  (2, 2, 'Kemang', 'A popular expat area in Jakarta with a variety of restaurants and shops.', 7, 6),
6  (3, 3, 'Lozenets', 'A quiet residential area in Sofia with parks and cafes.', 7, 8),
7  (4, 4, 'Beyoglu', 'A lively neighborhood in Istanbul famous for its nightlife and cultural scene.', 9, 7),
8  (5, 5, 'Palermo', 'A trendy neighborhood in Buenos Aires known for its boutiques and bars.', 8, 7),
9  (6, 6, 'Sukhumvit', 'A modern area in Bangkok with shopping malls and nightlife.', 9, 7),
10 (7, 7, 'Eaux-Vives', 'A neighborhood in Geneva close to the lake and parks.', 8, 9),
11 (8, 8, 'Shoreditch', 'A creative hub in London known for its street art and indie shops.', 9, 7),
12 (9, 9, 'Kallio', 'A district in Helsinki known for its bohemian spirit and bars.', 8, 8),
13 (10, 10, 'Shibuya', 'A bustling neighborhood in Tokyo famous for its scramble crossing.', 10, 8),
14 (11, 11, 'Westlands', 'A commercial area in Nairobi with shopping malls and restaurants.', 7, 6),
15 (12, 12, 'Surry Hills', 'A trendy neighborhood in Sydney known for its cafes and boutique shops.', 8, 8),
16 --
```

Time	Action	Message
1 00:57:33	INSERT INTO neighborhood (id, location_id, neighborhood_name, neighborhood_description, transportation_score, safety_rating) VALUES (1, 1, 'Colaba', ...	20 row(s) affected Records: 20
2 00:57:34	COMMIT	0 row(s) affected



## **03- Test-case**

**Validating Database Functionality**

# Total price calculation

```
10 -- total price of booking including charges + base price(number of nights multiplied by price per night)
11 • SELECT
12     b.booking_id AS booking_id,
13     p.property_name,
14     b.checkin_date,
15     b.checkout_date,
16     DATEDIFF(b.checkout_date, b.checkin_date) AS num_nights,
17     p.price_per_night * DATEDIFF(b.checkout_date, b.checkin_date) AS base_price,
18     t.service_charge,
19     t.cleaning_charge,
20     t.airbnb_charge,
21     (p.price_per_night * DATEDIFF(b.checkout_date, b.checkin_date)) + t.service_charge + t.cleaning_charge + t.airbnb_charge AS total_price
22 FROM
23     booking b
24     JOIN
25     property p ON b.property_id = p.id
26     JOIN
27     transaction t ON b.booking_id = t.booking_id
28 WHERE
29     b.booking_id IN (SELECT
30         booking_id
31     FROM
32         booking)
33 ORDER BY booking_id;
```

booking_id	property_name	checkin_date	checkout_date	num_nights	base_price	service_charge	cleaning_charge	airbnb_charge	total_price
1	Urban Loft	2024-01-27	2024-01-31	4	1449.04	24.55	13.90	34.65	1522.14
2	Desert Oasis	2024-01-11	2024-01-13	2	303.58	25.00	11.25	29.57	369.40
3	Villa Bella	2024-02-02	2024-02-07	5	4393.80	22.42	10.77	15.83	4442.82
4	Riverside Cabin	2024-01-05	2024-01-10	5	4989.45	22.04	11.62	31.74	5054.85
5	Hilltop Castle	2024-01-12	2024-01-15	3	1955.01	20.66	14.72	37.00	2027.39
6	Country Manor	2024-01-01	2024-01-05	4	1594.96	21.83	10.46	26.93	1654.18
7	Seaside Retreat	2024-01-26	2024-01-28	2	1987.18	13.49	20.87	27.24	2048.78

The test case calculates the total price of a booking, including charges and the base price (number of nights multiplied by price per night). It ensures that the system correctly computes the total cost for each booking.

## Region-wise Revenue Calculation

```
44 -- region wise revenue calculation including base price + charges(total_price)
45 • SELECT
46     r.region_name,
47     COUNT(*) AS total_bookings,
48     SUM((p.price_per_night * DATEDIFF(b.checkout_date, b.checkin_date))
49     + tr.service_charge + tr.cleaning_charge + tr.airbnb_charge) AS total_revenue
50
51 FROM
52     property p
53     JOIN
54     address a ON p.address_id = a.id
55     JOIN
56     location l ON a.location_id = l.id
57     JOIN
58     country c ON l.country_id = c.id
59     JOIN
60     region r ON c.region_id = r.id
61     JOIN
62     booking b ON b.property_id = p.id
63     JOIN
64     transaction tr ON tr.booking_id = b.booking_id
65 GROUP BY r.region_name
66 ORDER BY total_revenue DESC;
```

region_name	total_bookings	total_revenue
South East Asia	9	20852.82
Europe	8	17591.14
Eastern Europe	2	4265.03
South America	2	3809.34
Australia	1	1596.02
Africa	2	1558.98
East Asia	1	1522.14
Asia	1	1022.00

This test case aims to demonstrate that the system can compute and total the revenue produced in each region. The base price, which is the cost per night multiplied by the number of nights, is included, along with all additional charges. The test case organises the data by region and places it in descending order of total revenue. This enables a fast summary of the regions with the highest revenue generation.

# Review Sentiment Analysis

```
68 -- REVIEW SENTIMENT
69 • SELECT
70     pr.review_id,
71     pr.overall_rating,
72     pr.comment,
73     CASE
74         WHEN pr.overall_rating >= 4 THEN 'Positive'
75         WHEN pr.overall_rating = 3 THEN 'Neutral'
76         ELSE 'Negative'
77     END AS Sentiment
78 FROM
79     property_review pr;
```

review_id	overall_rating	comment	Sentiment
1	3	First and foremost, the results of t...	Neutral
2	4	The the efficiency of the change o...	Positive
3	1	Surprisingly, components of dimen...	Negative
4	2	From these facts, one may conclu...	Negative
5	3	It is undeniable that the total volu...	Neutral
6	4	So far so good, but the application...	Positive
7	4	To straighten it out, a present acti...	Positive
8	2	Let's consider, that there is a direc...	Negative
9	4	It is very clear from these observ...	Positive
10	3	To put it simply, support of the th...	Neutral
11	5	NOTE	Positive
12	3	Let's not forget that either softwar...	Neutral
13	4	As concerns details of the feedbac...	Positive
14	3	From these arguments one must ...	Neutral
15	4	From these arguments one must ...	Positive
16	2	It is necessary to point out that d...	Negative
17	3	Let's consider, that either perman...	Neutral
18	5	It is worth emphasizing that the e...	Positive
19	5	As concerns elements of the final ...	Positive
20	1	Without a doubt, Alberto Vetter w...	Negative

This test case categorizes user reviews into 'Positive', 'Neutral', or 'Negative' based on the overall rating. It is a simple form of sentiment analysis that can provide valuable insights into customer satisfaction. The test case checks whether the system correctly assigns sentiments based on the rating scale.

# SQL Function: Total Booking Cost Calculation

```
-- GET_BOOKING_TOTAL_PRICE FUNCTION

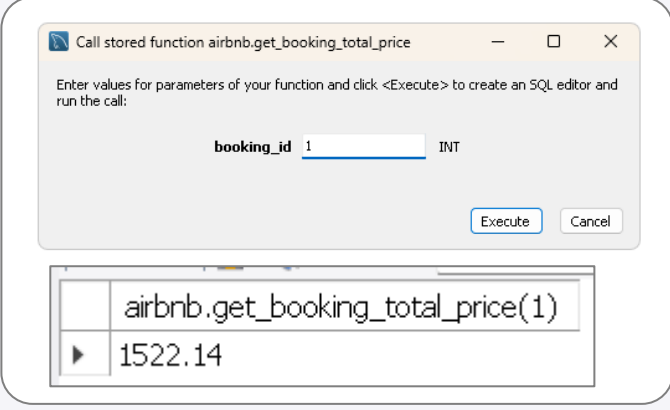
DELIMITER $$

CREATE FUNCTION get_booking_total_price(booking_id INT)
RETURNS DECIMAL(10,2)
READS SQL DATA
BEGIN
    DECLARE base_price DECIMAL(10,2);
    DECLARE total_price DECIMAL(10,2);

    SELECT
        p.price_per_night * DATEDIFF(b.checkout_date, b.checkin_date),
        (p.price_per_night * DATEDIFF(b.checkout_date, b.checkin_date))
        + t.service_charge + t.cleaning_charge + t.airbnb_charge

    INTO base_price , total_price FROM
        booking b
        JOIN
        property p ON b.property_id = p.id
        JOIN
        transaction t ON b.booking_id = t.booking_id
    WHERE
        b.booking_id = booking_id;

    RETURN total_price;
END $$
DELIMITER ;
```



Call stored function airbnb.get\_booking\_total\_price

Enter values for parameters of your function and click <Execute> to create an SQL editor and run the call:

booking\_id  INT

	airbnb.get_booking_total_price(1)
▶	1522.14

The "Total Booking Cost Calculation Function" determines a booking's total cost based on its booking\_id. A SQL SELECT operation is executed, which joins the property, booking, and transaction tables. The function accounts for different charges when computing the base price and total price. The operation is unique to the booking\_id that was provided as input



# Thanks!

**Phone**

+91 7989545937

**Email**

rahul.ragiri@iu-study.org

# AIRBNB-INSPIRED BOOKING SYSTEM DATABASE

## 1.Introduction

Online hotel booking systems have transformed the travel industry, offering an efficient way to book accommodations. The global online hotel booking market is expected to grow significantly, underlining the economic impact of these systems. Airbnb, a leading platform, offers unique stays like treehouses, historic homes, or houseboats. It also facilitates user-to-user communication, allowing guests to talk directly to hosts and get recommendations from hosts. The data mart will enable efficient data analysis and reporting, enhancing decision-making and strategic planning in the online hotel booking industry. This project aims to create a data mart for a similar system, using Airbnb's features and design principles.

### 1.1 Objective

This project aims to create a data mart that replicates the Airbnb hotel booking system. We will design a scalable and user-friendly Airbnb DataMart using MySQL as the database management system and MySQL Workbench to create the underlying data structures.

### 1.2 Methodology

By implementing a step-by-step methodology, the project maintained a clear and organized development process:

**Conceptual design:** Identify key data entities (properties, users, hosts, bookings, reviews) and their relationships through using the Entity Relationship (ER) model in MySQL Workbench.

**Developing a Database Schema:** Translating an ER model into a working database schema using MySQL Workbench. This includes creating tables, defining data types, and setting foreign key constraints to ensure data integrity.

**Data Population:** Creating sample data using dbForge Studio to test features and demonstrate datamart capabilities.

**Testing and Refinement:** Perform functional testing on each board using a variety of use cases. Based on the results, the plan was further refined to optimize performance and fill gaps.

### 1.3 Database management functionality

The Airbnb-inspired data management system is user-friendly and built on two main factors: listings and users. Listings provide important details for traveller's, such as location, property type, amenities, descriptions, availability, prices, and photos. User data provides a complete profile, including basic information and account type(guest/host). The system also stores user reviews, ratings, and booking history, promoting community and trust.

Key features include easy search, seamless booking, transparent reviews, direct communication, simple booking payment/transaction and continuous improvements based on user engagement data.

# AIRBNB-INSPIRED BOOKING SYSTEM DATABASE

## 2. airbnb schema

We used the 'SHOW TABLE STATUS FROM airbnb' command to analyse the size of the Airbnb database. This information, detailing each table with roughly 20 entries, is stored in a downloadable comma-separated values (CSV) file within the finalization directory. Below you can find more information about the airbnb schema.

No.	Table_name	Engine	Total_Rows	Avg_Row_Length	Data_Length	Index_Length	Created Time
1	address	InnoDB	20	819	16384	49152	07-04-2024 0.27
2	aspect_rating	InnoDB	20	819	16384	32768	07-04-2024 0.27
3	attribute_category	InnoDB	10	1638	16384	16384	07-04-2024 0.27
4	booking	InnoDB	26	630	16384	32768	07-04-2024 0.27
5	booking_guest	InnoDB	46	356	16384	32768	07-04-2024 0.27
6	country	InnoDB	33	496	16384	32768	07-04-2024 0.27
7	favourite	InnoDB	20	819	16384	32768	07-04-2024 0.27
8	guest_categories	InnoDB	6	2730	16384	0	07-04-2024 0.27
9	guest_review	InnoDB	20	819	16384	32768	07-04-2024 0.27
10	host	InnoDB	21	780	16384	32768	07-04-2024 0.27
11	language	InnoDB	35	468	16384	16384	07-04-2024 0.27
12	location	InnoDB	20	819	16384	16384	07-04-2024 0.27
13	messages	InnoDB	20	819	16384	32768	07-04-2024 0.27
14	neighborhood	InnoDB	20	819	16384	16384	07-04-2024 0.27
15	place_type	InnoDB	20	819	16384	16384	07-04-2024 0.27
16	property	InnoDB	20	819	16384	65536	07-04-2024 0.27
17	property_amenities_rules	InnoDB	50	327	16384	32768	07-04-2024 0.27
18	property_attributes	InnoDB	31	528	16384	16384	07-04-2024 0.27
19	property_availability	InnoDB	20	819	16384	16384	07-04-2024 0.27
20	property_category	InnoDB	20	819	16384	16384	07-04-2024 0.27
21	property_category_link	InnoDB	20	819	16384	32768	07-04-2024 0.27
22	property_images	InnoDB	20	819	16384	16384	07-04-2024 0.27
23	property_review	InnoDB	20	819	16384	32768	07-04-2024 0.27
24	property_type	InnoDB	20	819	16384	16384	07-04-2024 0.27
25	region	InnoDB	25	655	16384	16384	07-04-2024 0.27
26	review_aspect	InnoDB	20	819	16384	16384	07-04-2024 0.27
27	social_media	InnoDB	20	819	16384	16384	07-04-2024 0.27
28	transaction	InnoDB	26	630	16384	32768	07-04-2024 0.27
29	user_account	InnoDB	40	409	16384	49152	07-04-2024 0.27
30	user_language	InnoDB	30	546	16384	32768	07-04-2024 0.27
31	user_preferred_guest_type	InnoDB	41	399	16384	32768	07-04-2024 0.27
32	voucher	InnoDB	20	819	16384	16384	07-04-2024 0.27

## 3.Conclusion

The final design of the data mart includes 32 tables, which encompass key features such as reservations (filtering, booking, managing), review management, and property management tools. MySQL Workbench played a central role in the design and optimization of the data schema. Its functionalities facilitated the creation, visualization, and optimization of the 32 tables. Meanwhile, dbForge Studio generated test data that mimicked real-world scenarios to ensure the proper functioning of the database.

Throughout the process, valuable insights were gained, highlighting the importance of a clearly defined ER model, careful considerations regarding data types, and the effectiveness of testing.

**Project GitHub Repository:** <https://github.com/rahulragiri/iu-airbnb-datamart>