

The background image shows a spiral-bound notebook on a wooden desk. The notebook is filled with hand-drawn sketches of user interface elements for an Airbnb application. At the top, the word "#Content" is written in large, bold, black letters. Below it, there are several sketches of different screens or components. One sketch shows a list of items with a search bar and a filter button. Another sketch shows a detailed view of a listing with a large image and text. A third sketch shows a map view with a list of locations. The sketches are drawn with black ink and some green and red highlights. A hand holding a black pen is visible in the bottom right corner, pointing at one of the sketches. A white keyboard is partially visible in the top right corner.

SQL PROJECT: Database Design and Implementation for Airbnb Use Case

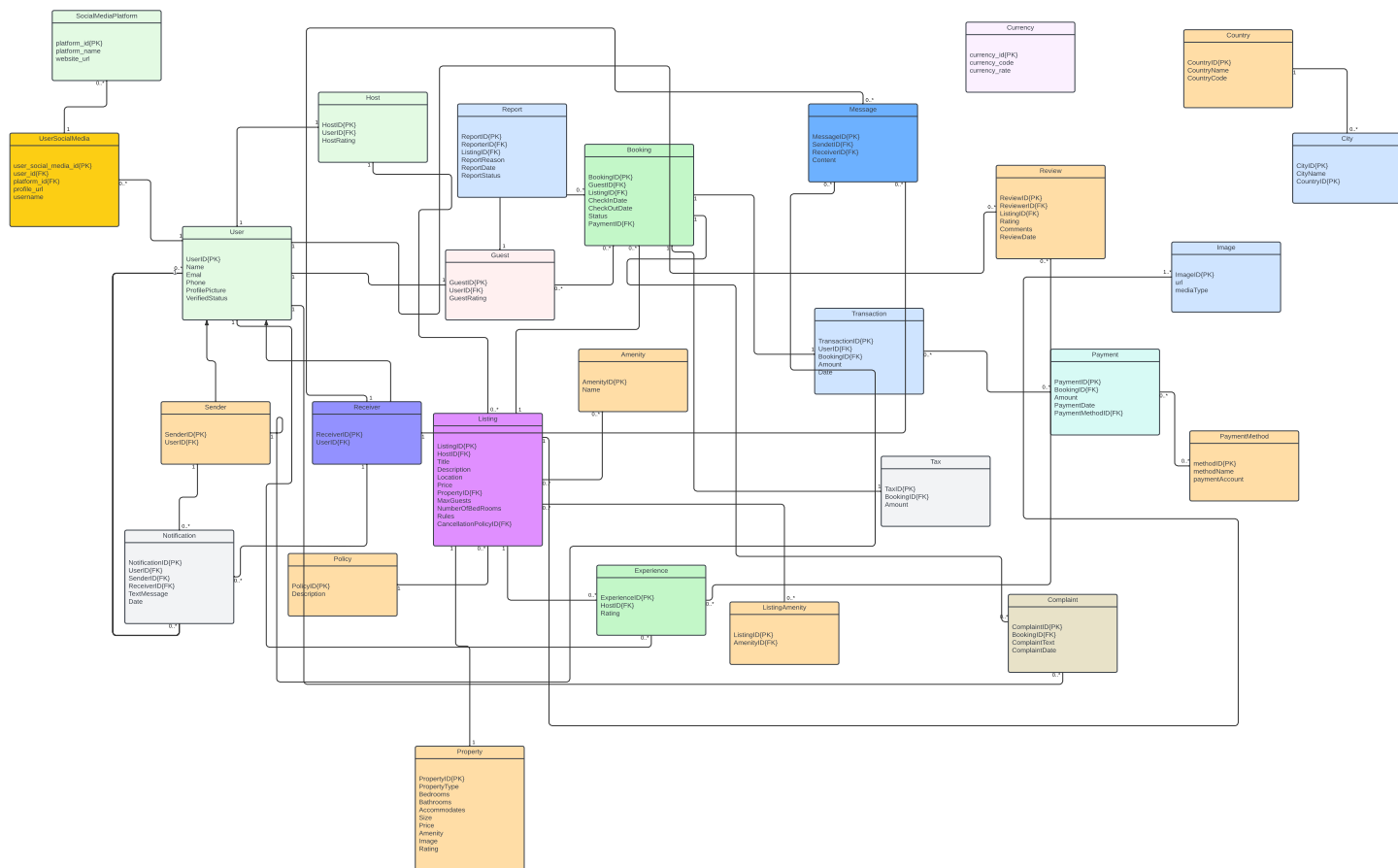
SALAHEDDINE SAMID

Matriculation: 3220618

IU International University of applied sciences

SUMMARY

- This project aim to design and develop a relational database for the online booking platform(Airbnb). The project was structures across different steps: Identify entities, define relationships, determine primary keys, conversion into physical data model, define and normalize attributes, determine consistency conditions, and formulate transactions.



ER Model

Data dictionary

Comments	STRING	Review comment
ReviewDate	DATE	Date of the review

ReportID	INTEGER	Unique identifier for the report
ReporterID	INTEGER	Foreign key to the reporter
ListingID	INTEGER	Foreign key to listing
ReportReason	STRING	Reason for reporting
ReportDate	DATE	Date for reporting
ReportStatus	STRING	Status of the report

MessageID	INTEGER	Unique identifier for the message
SenderID	INTEGER	Foreign key to the user
ReceiverID	INTEGER	Foreign key to user
Content	STRING	Content of the message

NotificationID	INTEGER	Unique identifier for the notification
UserID	INTEGER	Foreign key to user
SenderID	INTEGER	Foreign key to the sender
ReceiverID	INTEGER	Foreign key to the receiver
TextMessage	STRING	Notification text
Date	DATE	Date of the notification

PolicyID	INTEGER	Unique identifier for the policy
Description	STRING	Description of the policy

AmenityID	INTEGER	Description
Name	STRING	Name of the amenity

ListingID	INTEGER	Foreign key to the listing
AmenityID	INTEGER	Foreign key to the amenity

ExperienceID	INTEGER	Unique identifier for the experience
--------------	---------	--------------------------------------

BookingID	INTEGER	Unique identifier for the booking
GuestID	INTEGER	Foreign key to guest
ListingID	INTEGER	Foreign key to listing
CheckInDate	DATE	Check-in date
CheckOutDate	DATE	Check-out date
Status	STRING	Status of the booking
PaymentID	INTEGER	Foreign key to payment

PaymentID	INTEGER	Unique identifier for the payment
BookingID	INTEGER	Foreign key to booking
Amount	Float	Amount paid
PaymentDate	DATE	Date of payment
PaymentMethodID	INTEGER	Foreign key to payment method

MethodID	INTEGER	Unique identifier for the method
MethodName	STRING	Name of the payment
PaymentAccount	STRING	Account used for payment

TransactionID	INTEGER	Unique identifier for the transaction
UserID	INTEGER	Foreign key to user
BookingID	INTEGER	Foreign key to booking
Amount	FLOAT	Transaction amount
Date	DATE	Transaction date

TaxID	INTEGER	Unique identifier
BookingID	INTEGER	Foreign key to booking
Amount	FLOAT	Tax amount

ReviewID	INTEGER	Unique identifier for the review
ReviewerID	INTEGER	Foreign key to the user
ListingID	INTEGER	Foreign key to listing
Rating	FLOAT	Rating given

HostID	INTEGER	Foreign key to the host
Rating	FLOAT	Rating of the experience

ImageID	INTEGER	Unique identifier for the image
URL	STRING	URL to the image
MediaType	STRING	Type of the media

ComplaintID	INTEGER	Unique identifier for the complaint
BookingID	INTEGER	Foreign key to booking
ComplaintText	STRING	Text of the complaint
ComplaintDate	DATE	Date of complaint

User:

UserID	INTEGER	Unique identifier for the user
Name	STRING	Name of the user
Email	STRING	Email address of the user
Phone	STRING	Phone number of the user
ProfilePicture	STRING	URL of the profile picture
VerifiedStatus	BOOLEAN	Verification status of the user

GuestID	INTEGER	Unique identifier for the host
UserID	INTEGER	Foreign key to user
GuestRating	FLOAT	Rating of the guest

HostID	INTEGER	Unique identifier for the host
UserID	INTEGER	Foreign key to user
HostRating	FLOAT	Rating of the guest

ListingID	INTEGER	Unique identifier for the listing
HostID	INTEGER	Foreign key to host
Title	STRING	Title of the listing
Description	STRING	Description of the listing
Location	STRING	Location of the listing
Price	FLOAT	Price per night
PropertyID	INTEGER	Type of the property (Foreign key)
MaxGuests	INTEGER	Maximum number of guests allowed
NumberOfBedRooms	INTEGER	Number of bedrooms
Rules	STRING	Rules for the listing
CancellationPolicyID	INTEGER	Foreign key to policy.

User table implementation

Create User table:

```
CREATE TABLE User(  
  UserID INTEGER NOT NULL PRIMARY KEY,  
  Name VARCHAR(200),  
  Email VARCHAR(100),  
  Phone VARCHAR(100),  
  ProfilePicture VARCHAR(250),  
  VerifiedStatus BOOLEAN  
);
```

Field	Type	Null	Key	Default	Extra
UserID	int	NO	PRI	NULL	
Name	varchar(200)	YES		NULL	
Email	varchar(100)	YES		NULL	
Phone	varchar(100)	YES		NULL	
profile_picture	varchar(250)	YES		NULL	
verified_status	tinyint(1)	YES		NULL	

6 rows in set (0.00 sec)

Insert User values:

```
mysql> development> insert_values > user.sql  
INSERT INTO User (UserID, Name, Email, Phone, profile_picture, verified_status) VALUES  
(1, 'Alice Smith', 'alice.smith@example.com', '123-456-7890', 'https://example.com/profiles/alice.jpg', TRUE),  
(2, 'Bob Johnson', 'bob.johnson@example.com', '980-765-4321', 'https://example.com/profiles/bob.jpg', FALSE),  
(3, 'Charlie Brown', 'charlie.brown@example.com', '555-555-5555', 'https://example.com/profiles/charlie.jpg', TRUE),  
(4, 'Dana White', 'dana.white@example.com', '444-444-4444', 'https://example.com/profiles/dana.jpg', FALSE),  
(5, 'Eve Black', 'eve.black@example.com', '333-333-3333', 'https://example.com/profiles/eve.jpg', TRUE),  
(6, 'Frank Adams', 'frank.adams@example.com', '222-222-2222', 'https://example.com/profiles/frank.jpg', FALSE),  
(7, 'Grace Taylor', 'grace.taylor@example.com', '777-777-7777', 'https://example.com/profiles/grace.jpg', TRUE),  
(8, 'Henry Lee', 'henry.lee@example.com', '666-666-6666', 'https://example.com/profiles/henry.jpg', FALSE),  
(9, 'Ivy Clark', 'ivy.clark@example.com', '999-999-9999', 'https://example.com/profiles/ivy.jpg', TRUE),  
(10, 'Jack Harris', 'jack.harris@example.com', '111-111-1111', 'https://example.com/profiles/jack.jpg', FALSE),  
(11, 'Kate Miller', 'kate.miller@example.com', '444-555-6666', 'https://example.com/profiles/kate.jpg', TRUE),  
(12, 'Luke Evans', 'luke.evans@example.com', '777-888-9999', 'https://example.com/profiles/luke.jpg', FALSE),  
(13, 'Mia Roberts', 'mia.roberts@example.com', '222-333-4444', 'https://example.com/profiles/mia.jpg', TRUE),  
(14, 'Nick Wilson', 'nick.wilson@example.com', '555-666-7777', 'https://example.com/profiles/nick.jpg', FALSE),  
(15, 'Olivia Brown', 'olivia.brown@example.com', '888-999-0000', 'https://example.com/profiles/olivia.jpg', TRUE),  
(16, 'Paul Davis', 'paul.davis@example.com', '111-222-3333', 'https://example.com/profiles/paul.jpg', FALSE),  
(17, 'Quinn Allen', 'quinn.allen@example.com', '333-444-5555', 'https://example.com/profiles/quinn.jpg', TRUE),  
(18, 'Rachel Young', 'rachel.young@example.com', '666-777-8888', 'https://example.com/profiles/rachel.jpg', FALSE),  
(19, 'Samuel Hill', 'samuel.hill@example.com', '999-000-1111', 'https://example.com/profiles/samuel.jpg', TRUE),  
(20, 'Tina Carter', 'tina.carter@example.com', '222-333-4444', 'https://example.com/profiles/tina.jpg', FALSE);
```

Database changed

```
mysql> SELECT * FROM User;
```

UserID	Name	Email	Phone	profile_picture	verified_status
1	Alice Smith	alice.smith@example.com	123-456-7890	default_profile_picture_url.jpg	0
2	Bob Johnson	bob.johnson@example.com	980-765-4321	default_profile_picture_url.jpg	0
3	Charlie Brown	charlie.brown@example.com	555-555-5555	default_profile_picture_url.jpg	0
4	Dana White	dana.white@example.com	444-444-4444	default_profile_picture_url.jpg	0
5	Eve Black	eve.black@example.com	333-333-3333	default_profile_picture_url.jpg	0
6	Frank Adams	frank.adams@example.com	222-222-2222	default_profile_picture_url.jpg	0
7	Grace Taylor	grace.taylor@example.com	777-777-7777	default_profile_picture_url.jpg	0
8	Henry Lee	henry.lee@example.com	666-666-6666	default_profile_picture_url.jpg	0
9	Ivy Clark	ivy.clark@example.com	999-999-9999	default_profile_picture_url.jpg	0
10	Jack Harris	jack.harris@example.com	111-111-1111	default_profile_picture_url.jpg	0
11	Kate Miller	kate.miller@example.com	444-555-6666	default_profile_picture_url.jpg	0
12	Luke Evans	luke.evans@example.com	777-888-9999	default_profile_picture_url.jpg	0
13	Mia Roberts	mia.roberts@example.com	222-333-4444	default_profile_picture_url.jpg	0
14	Nick Wilson	nick.wilson@example.com	555-666-7777	default_profile_picture_url.jpg	0
15	Olivia Brown	olivia.brown@example.com	888-999-0000	default_profile_picture_url.jpg	0
16	Paul Davis	paul.davis@example.com	111-222-3333	default_profile_picture_url.jpg	0
17	Quinn Allen	quinn.allen@example.com	333-444-5555	default_profile_picture_url.jpg	0
18	Rachel Young	rachel.young@example.com	666-777-8888	default_profile_picture_url.jpg	0
19	Samuel Hill	samuel.hill@example.com	999-000-1111	default_profile_picture_url.jpg	0
20	Tina Carter	tina.carter@example.com	222-333-4444	default_profile_picture_url.jpg	0

20 rows in set (0.00 sec)

Host table implementation

Create Host table:

```
CREATE TABLE HOST (  
  HostID INTEGER NOT NULL PRIMARY KEY,  
  UserID INTEGER,  
  HostRating FLOAT,  
  FOREIGN KEY (UserID) REFERENCES User (UserID)  
);
```

```
mysql> DESC Host;
```

Field	Type	Null	Key	Default	Extra
HostID	int	NO	PRI	NULL	
UserID	int	YES	MUL	NULL	
HostRating	float	YES		NULL	

3 rows in set (0.00 sec)

Insert Host values:

```
INSERT INTO HOST (HostID, UserID, HostRating) VALUES  
(1, 1, 4.5),  
(2, 2, 4.0),  
(3, 3, 3.5),  
(4, 4, 4.2),  
(5, 5, 4.8),  
(6, 6, 4.7),  
(7, 7, 4.3),  
(8, 8, 4.6),  
(9, 9, 4.1),  
(10, 10, 3.9),  
(11, 11, 4.4),  
(12, 12, 4.0),  
(13, 13, 3.8),  
(14, 14, 4.0),  
(15, 15, 4.3),  
(16, 16, 4.2),  
(17, 17, 4.7),  
(18, 18, 3.6),  
(19, 19, 4.5),  
(20, 20, 4.8);
```

```
mysql> SELECT * FROM Host;
```

HostID	UserID	HostRating
1	1	4.5
2	2	4
3	3	3.5
4	4	4.2
5	5	4.8
6	6	4.7
7	7	4.3
8	8	4.6
9	9	4.1
10	10	3.9
11	11	4.4
12	12	4.0
13	13	3.8
14	14	4
15	15	4.3
16	16	4.2
17	17	4.7
18	18	3.6
19	19	4.5
20	20	4.8

20 rows in set (0.01 sec)

Guest table implementation

Create Guest table:

```
CREATE TABLE Guest(  
  guestID INTEGER NOT NULL PRIMARY KEY,  
  UserID INTEGER,  
  GuestRating FLOAT,  
  FOREIGN KEY(UserID) REFERENCES USER (UserID)  
);
```

```
mysql> DESC Guest;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| guestID | int | NO | PRI | NULL | |  
| UserID | int | YES | MUL | NULL | |  
| GuestRating | float | YES | | NULL | |  
+-----+-----+-----+-----+-----+-----+  
3 rows in set (0.00 sec)
```

Insert Guest values:

```
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (1, 1, 4.5);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (2, 2, 4);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (3, 3, 3.5);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (4, 4, 4.2);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (5, 5, 4.8);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (6, 6, 4.7);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (7, 7, 4.3);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (8, 8, 4.6);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (9, 9, 4.1);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (10, 10, 3.9);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (11, 11, 4.4);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (12, 12, 4.9);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (13, 13, 3.8);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (14, 14, 4);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (15, 15, 4.3);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (16, 16, 4.2);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (17, 17, 4.7);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (18, 18, 3.6);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (19, 19, 4.5);  
INSERT INTO Guest (GuestID, UserID, GuestRating) VALUES (20, 20, 4.8);
```

```
mysql> SELECT * FROM Guest  
+-----+-----+-----+  
| guestID | UserID | GuestRating |  
+-----+-----+-----+  
| 1 | 1 | 4.5 |  
| 2 | 2 | 4 |  
| 3 | 3 | 3.5 |  
| 4 | 4 | 4.2 |  
| 5 | 5 | 4.8 |  
| 6 | 6 | 4.7 |  
| 7 | 7 | 4.3 |  
| 8 | 8 | 4.6 |  
| 9 | 9 | 4.1 |  
| 10 | 10 | 3.9 |  
| 11 | 11 | 4.4 |  
| 12 | 12 | 4.9 |  
| 13 | 13 | 3.8 |  
| 14 | 14 | 4 |  
| 15 | 15 | 4.3 |  
| 16 | 16 | 4.2 |  
| 17 | 17 | 4.7 |  
| 18 | 18 | 3.6 |  
| 19 | 19 | 4.5 |  
| 20 | 20 | 4.8 |  
+-----+-----+-----+  
20 rows in set (0.00 sec)
```

Property table implementation

Create Property table:

```
CREATE TABLE Property (
  PropertyID INTEGER NOT NULL PRIMARY KEY,
  PropertyType VARCHAR(100),
  Bedrooms INTEGER,
  Bathrooms INTEGER,
  Accommodates INTEGER,
  Size FLOAT,
  Amenities VARCHAR(200),
  ImageURL VARCHAR(200),
  Rating FLOAT
);
```

```
mysql> DESC Property;
```

Field	Type	Null	Key	Default	Extra
PropertyID	int	NO	PRI	NONE	
PropertyType	varchar(100)	YES		NONE	
Bedrooms	int	YES		NONE	
Bathrooms	int	YES		NONE	
Accommodates	int	YES		NONE	
Size	float	YES		NONE	
Amenities	varchar(200)	YES		NONE	
ImageURL	varchar(200)	YES		NONE	
Rating	float	YES		NONE	

```
9 rows in set (0.01 sec)
```

Insert Property values:

```
INSERT INTO Property (PropertyID, PropertyType, Bedrooms, Bathrooms, Accommodates, Size, Amenities, ImageURL, Rating) VALUES
(1, 'Apartment', 2, 1, 4, 80.5, 'Wi-Fi, Kitchen, Parking', 'http://example.com/apartment1.jpg', 4.5),
(2, 'House', 3, 2, 6, 150.0, 'Garden, Pool, BBQ', 'http://example.com/house1.jpg', 4.8),
(3, 'Villa', 4, 3, 8, 300.0, 'Private beach, Jacuzzi, Tennis court', 'http://example.com/villa1.jpg', 4.9),
(4, 'Cabin', 1, 1, 2, 60.0, 'Fireplace, Mountain view, Pet-friendly', 'http://example.com/cabin1.jpg', 4.2),
(5, 'Studio', 0, 1, 2, 40.0, 'Central location, Gym access', 'http://example.com/studio1.jpg', 4.0),
(6, 'Loft', 1, 1, 3, 70.0, 'City view, Modern design', 'http://example.com/loft1.jpg', 4.7),
(7, 'Farmhouse', 5, 3, 10, 250.0, 'Farm experience, Family-friendly', 'http://example.com/farmhouse1.jpg', 4.6),
(8, 'Chalet', 2, 2, 4, 100.0, 'Ski-in/ski-out, Hot tub', 'http://example.com/chalet1.jpg', 4.4),
(9, 'Beach House', 4, 2, 8, 200.0, 'Oceanfront, Deck with sea view', 'http://example.com/beachhouse1.jpg', 4.9),
(10, 'Penthouse', 3, 3, 6, 100.0, 'Luxury amenities, Rooftop terrace', 'http://example.com/penthouse1.jpg', 4.8),
(11, 'Bungalow', 2, 1, 4, 90.0, 'Private garden, BBQ area', 'http://example.com/bungalow1.jpg', 4.3),
(12, 'Castle', 10, 5, 20, 1000.0, 'Historic property, Ballroom', 'http://example.com/castle1.jpg', 4.7),
(13, 'Treehouse', 1, 0, 2, 25.0, 'Nature retreat, Secluded', 'http://example.com/treehouse1.jpg', 4.5),
(14, 'Yacht', 3, 2, 6, 150.0, 'Boat experience, Captain included', 'http://example.com/yacht1.jpg', 4.6),
(15, 'Mansion', 8, 6, 16, 500.0, 'Grand estate, Swimming pool', 'http://example.com/mansion1.jpg', 4.9),
(16, 'Cottage', 2, 1, 4, 80.0, 'Quaint charm, Countryside', 'http://example.com/cottage1.jpg', 4.2),
(17, 'Townhouse', 3, 2, 6, 120.0, 'City living, Walkable location', 'http://example.com/townhouse1.jpg', 4.1),
(18, 'Guesthouse', 5, 3, 10, 200.0, 'Hosted experience, Local tips', 'http://example.com/guesthouse1.jpg', 4.3),
(19, 'Apartment', 1, 1, 2, 50.0, 'Downtown, Modern amenities', 'http://example.com/apartment2.jpg', 4.0),
(20, 'Farm Stay', 3, 2, 6, 120.0, 'Farm life, Fresh produce', 'http://example.com/farmstay1.jpg', 4.5);
```

```
mysql> SELECT * FROM property;
```

PropertyID	PropertyType	Bedrooms	Bathrooms	Accommodates	Size	Amenities	ImageURL	Rating
1	Apartment	2	1	4	80.5	Wi-Fi, Kitchen, Parking	http://example.com/apartment1.jpg	4.5
2	House	3	2	6	150	Garden, Pool, BBQ	http://example.com/house1.jpg	4.8
3	Villa	4	3	8	300	Private beach, Jacuzzi, Tennis court	http://example.com/villa1.jpg	4.9
4	Cabin	1	1	2	60	Fireplace, Mountain view, Pet-friendly	http://example.com/cabin1.jpg	4.2
5	Studio	0	1	2	40	Central location, Gym access	http://example.com/studio1.jpg	4.0
6	Loft	1	1	3	70	City view, Modern design	http://example.com/loft1.jpg	4.7
7	Farmhouse	5	3	10	250	Farm experience, Family-friendly	http://example.com/farmhouse1.jpg	4.6
8	Chalet	2	2	4	100	Ski-in/ski-out, Hot tub	http://example.com/chalet1.jpg	4.4
9	Beach House	4	2	8	200	Oceanfront, Deck with sea view	http://example.com/beachhouse1.jpg	4.9
10	Penthouse	3	3	6	100	Luxury amenities, Rooftop terrace	http://example.com/penthouse1.jpg	4.8
11	Bungalow	2	1	4	90	Private garden, BBQ area	http://example.com/bungalow1.jpg	4.3
12	Castle	10	5	20	1000	Historic property, Ballroom	http://example.com/castle1.jpg	4.7
13	Treehouse	1	0	2	25	Nature retreat, Secluded	http://example.com/treehouse1.jpg	4.5
14	Yacht	3	2	6	150	Boat experience, Captain included	http://example.com/yacht1.jpg	4.6
15	Mansion	8	6	16	500	Grand estate, Swimming pool	http://example.com/mansion1.jpg	4.9
16	Cottage	2	1	4	80	Quaint charm, Countryside	http://example.com/cottage1.jpg	4.2
17	Townhouse	3	2	6	120	City living, Walkable location	http://example.com/townhouse1.jpg	4.1
18	Guesthouse	5	3	10	200	Hosted experience, Local tips	http://example.com/guesthouse1.jpg	4.3
19	Apartment	1	1	2	50	Downtown, Modern amenities	http://example.com/apartment2.jpg	4.0
20	Farm Stay	3	2	6	120	Farm life, Fresh produce	http://example.com/farmstay1.jpg	4.5

```
20 rows in set (0.00 sec)
```


Listing table implementation

Create Listing table:

```
CREATE TABLE Listing (
  ListingID INTEGER NOT NULL PRIMARY KEY,
  HostID INTEGER,
  Title VARCHAR(200),
  Description VARCHAR(200),
  Location VARCHAR(200),
  Price FLOAT,
  PropertyID INTEGER,
  MaxGuests INTEGER,
  NumberOfGuests INTEGER,
  Rules VARCHAR(200),
  CancellationPolicyID INTEGER,
  FOREIGN KEY (HostID) REFERENCES Host(HostID),
  FOREIGN KEY (CancellationPolicyID) REFERENCES Policy(policyID),
  FOREIGN KEY (PropertyID) REFERENCES Property(PropertyID)
);
```

Insert Listing values:

```
INSERT INTO Listing (ListingID, HostID, Title, Description, Location, Price, PropertyID, MaxGuests, NumberOfGuests, Rules, CancellationPolicyID)
VALUES
(1, 1, 'Cozy Apartment in Downtown', 'Modern and cozy apartment with city view', 'Downtown', 100.00, 1, 4, 2, 'No smoking, No pets allowed', 1),
(2, 2, 'Luxury Villa with Private Pool', 'Spacious villa with panoramic ocean views', 'Oceanfront', 500.00, 3, 8, 4, 'Pets allowed with extra charge', 2),
(3, 3, 'Charming Cottage in the Countryside', 'Quaint cottage surrounded by nature', 'Countryside', 80.00, 5, 2, 2, 'Quiet hours after 10 PM', 3),
(4, 4, 'Modern Loft in the City Center', 'Stylish loft with rooftop terrace', 'City Center', 150.00, 6, 3, 2, 'No parties or events', 4),
(5, 5, 'Secluded Cabin in the Woods', 'Remote cabin with fireplace and mountain view', 'Forest', 120.00, 4, 3, 1, 'No smoking indoors', 5),
(6, 1, 'Beachfront Bungalow with BBQ Area', 'Cozy bungalow steps away from the beach', 'Beachfront', 200.00, 7, 4, 3, 'BBQ area available', 6),
(7, 2, 'Historic Castle with Ballroom', 'Grand castle with historic charm', 'Countryside', 1000.00, 12, 20, 10, 'Respect the historic property', 7),
(8, 3, 'Rustic Farmhouse with Farm Experience', 'Family-friendly farmhouse with farm activities', 'Rural', 180.00, 11, 6, 4, 'Enjoy farm-fresh produce', 8),
(9, 4, 'Modern Townhouse with City View', 'Townhouse with modern amenities and city views', 'City Center', 250.00, 17, 6, 4, 'No pets allowed', 9),
(10, 5, 'Spacious Guesthouse with Local Tips', 'Welcoming guesthouse with local hospitality', 'Suburb', 120.00, 18, 8, 5, 'Explore nearby attractions', 10),
(11, 1, 'Ski-in/Ski-out Chalet with Hot Tub', 'Chalet perfect for ski enthusiasts', 'Mountain', 300.00, 8, 6, 3, 'Hot tub available for extra charge', 11),
(12, 2, 'Modern Penthouse with Luxury Amenities', 'Luxurious penthouse with rooftop terrace', 'Downtown', 400.00, 10, 4, 2, 'No smoking, No parties', 12),
(13, 3, 'Peaceful Treehouse Retreat', 'Secluded treehouse getaway in nature', 'Forest', 150.00, 13, 2, 1, 'Access via ladder, Not suitable for small children', 13),
(14, 4, 'Yacht Experience with Captain', 'Luxury yacht with onboard captain', 'Marina', 600.00, 14, 8, 6, 'Boat rules apply, No pets allowed', 14),
(15, 5, 'Grand Mansion with Swimming Pool', 'Elegant mansion with spacious grounds', 'Suburb', 800.00, 15, 16, 10, 'Pool rules must be followed', 15),
(16, 1, 'Urban Studio with Gym Access', 'Modern studio apartment with gym facilities', 'City Center', 80.00, 5, 2, 1, 'Quiet hours after 10 PM', 16),
(17, 2, 'Family-Friendly Apartment with Playground', 'Apartment ideal for families with playground', 'Suburb', 120.00, 10, 6, 4, 'Children must be supervised', 17),
(18, 3, 'Mountain View Cabin Retreat', 'Cozy cabin with stunning mountain views', 'Mountain', 100.00, 16, 3, 2, 'No smoking, No pets allowed', 18),
(19, 4, 'Luxury Beach House with Deck', 'Beach house with oceanfront deck', 'Beachfront', 350.00, 9, 8, 6, 'Deck access restricted after 10 PM', 19),
(20, 5, 'Farm Stay Experience with Fresh Produce', 'Farm stay with agricultural activities', 'Rural', 150.00, 20, 4, 2, 'Explore the farm', 20);
```

```
mysql> desc listing;
```

Field	Type	Null	Key	Default	Extra
ListingID	int	NO	PRI	NULL	
HostID	int	YES	MUL	NULL	
Title	varchar(200)	YES		NULL	
Description	varchar(200)	YES		NULL	
Location	varchar(200)	YES		NULL	
Price	float	YES		NULL	
PropertyID	int	YES	MUL	NULL	
MaxGuests	int	YES		NULL	
NumberOfGuests	int	YES		NULL	
Rules	varchar(200)	YES		NULL	
CancellationPolicyID	int	YES	MUL	NULL	

11 rows in set (0.00 sec)

```
mysql> SELECT * FROM Listing;
```

ListingID	HostID	Title	Description	Location	Price	PropertyID	MaxGuests	NumberOfGuests	Rules	CancellationPolicyID
1	1	Cozy Apartment in Downtown	Modern and cozy apartment with city view	Downtown	100	1	4	2	No smoking, No pets allowed	1
2	2	Luxury Villa with Private Pool	Spacious villa with panoramic ocean views	Oceanfront	500	3	8	4	Pets allowed with extra charge	2
3	3	Charming Cottage in the Countryside	Quaint cottage surrounded by nature	Countryside	80	5	2	2	Quiet hours after 10 PM	3
4	4	Modern Loft in the City Center	Stylish loft with rooftop terrace	City Center	150	6	3	2	No parties or events	4
5	5	Secluded Cabin in the Woods	Remote cabin with fireplace and mountain view	Forest	120	4	3	1	No smoking indoors	5
6	1	Beachfront Bungalow with BBQ Area	Cozy bungalow steps away from the beach	Beachfront	200	7	4	3	BBQ area available	6
7	2	Historic Castle with Ballroom	Grand castle with historic charm	Countryside	1000	12	20	10	Respect the historic property	7
8	3	Rustic Farmhouse with Farm Experience	Family-friendly farmhouse with farm activities	Rural	180	11	6	4	Enjoy farm-fresh produce	8
9	4	Modern Townhouse with City View	Townhouse with modern amenities and city views	City Center	250	17	6	4	No pets allowed	9
10	5	Spacious Guesthouse with Local Tips	Welcoming guesthouse with local hospitality	Suburb	120	18	8	5	Explore nearby attractions	10
11	1	Ski-in/Ski-out Chalet with Hot Tub	Chalet perfect for ski enthusiasts	Mountain	300	8	6	3	Hot tub available for extra charge	11
12	2	Modern Penthouse with Luxury Amenities	Luxurious penthouse with rooftop terrace	Downtown	400	10	4	2	No smoking, No parties	12
13	3	Peaceful Treehouse Retreat	Secluded treehouse getaway in nature	Forest	150	13	2	1	Access via ladder, Not suitable for small children	13
14	4	Yacht Experience with Captain	Luxury yacht with onboard captain	Marina	600	14	8	6	Boat rules apply, No pets allowed	14
15	5	Grand Mansion with Swimming Pool	Elegant mansion with spacious grounds	Suburb	800	15	16	10	Pool rules must be followed	15
16	1	Urban Studio with Gym Access	Modern studio apartment with gym facilities	City Center	80	5	2	1	Quiet hours after 10 PM	16
17	2	Family-Friendly Apartment with Playground	Apartment ideal for families with playground	Suburb	120	10	6	4	Children must be supervised	17
18	3	Mountain View Cabin Retreat	Cozy cabin with stunning mountain views	Mountain	100	16	3	2	No smoking, No pets allowed	18
19	4	Luxury Beach House with Deck	Beach house with oceanfront deck	Beachfront	350	9	8	6	Deck access restricted after 10 PM	19
20	5	Farm Stay Experience with Fresh Produce	Farm stay with agricultural activities	Rural	150	20	4	2	Explore the farm	20

20 rows in set (0.00 sec)

Booking table implementation

Create Booking table:

```
CREATE TABLE Booking (  
  BookingID INT(11) PRIMARY KEY,  
  GuestID INT(11),  
  ListingID INT(11),  
  CheckInDate DATE,  
  CheckOutDate DATE,  
  PaymentID INT(11),  
  Status VARCHAR(100),  
  FOREIGN KEY (GuestID) REFERENCES Guest(GuestID),  
  FOREIGN KEY (ListingID) REFERENCES Listing(ListingID),  
  FOREIGN KEY (PaymentID) REFERENCES Payment(PaymentID)  
)
```

```
mysql> desc booking;
```

Field	Type	Null	Key	Default	Extra
BookingID	int	NO	PK	NULL	
GuestID	int	YES	MUL	NULL	
ListingID	int	YES	MUL	NULL	
CheckInDate	date	YES		NULL	
CheckOutDate	date	YES		NULL	
PaymentID	int	YES	MUL	NULL	
Status	varchar(100)	YES		NULL	

7 rows in set (0.00 sec)

Insert Booking values:

```
INSERT INTO Booking (BookingID, GuestID, ListingID, CheckInDate, CheckOutDate, Status, PaymentID)  
VALUES  
(1, 1, '2024-07-01', '2024-07-05', 'Confirmed', 1),  
(2, 2, '2024-07-02', '2024-07-06', 'Confirmed', 2),  
(3, 3, '2024-07-03', '2024-07-07', 'Confirmed', 3),  
(4, 4, '2024-07-04', '2024-07-08', 'Confirmed', 4),  
(5, 5, '2024-07-05', '2024-07-09', 'Confirmed', 5),  
(6, 1, 6, '2024-07-06', '2024-07-10', 'Confirmed', 6),  
(7, 2, 7, '2024-07-07', '2024-07-11', 'Confirmed', 7),  
(8, 3, 8, '2024-07-08', '2024-07-12', 'Confirmed', 8),  
(9, 4, 9, '2024-07-09', '2024-07-13', 'Confirmed', 9),  
(10, 5, 10, '2024-07-10', '2024-07-14', 'Confirmed', 10),  
(11, 1, 11, '2024-07-11', '2024-07-15', 'Confirmed', 11),  
(12, 2, 12, '2024-07-12', '2024-07-16', 'Confirmed', 12),  
(13, 3, 13, '2024-07-13', '2024-07-17', 'Confirmed', 13),  
(14, 4, 14, '2024-07-14', '2024-07-18', 'Confirmed', 14),  
(15, 5, 15, '2024-07-15', '2024-07-19', 'Confirmed', 15),  
(16, 1, 16, '2024-07-16', '2024-07-20', 'Confirmed', 16),  
(17, 2, 17, '2024-07-17', '2024-07-21', 'Confirmed', 17),  
(18, 3, 18, '2024-07-18', '2024-07-22', 'Confirmed', 18),  
(19, 4, 19, '2024-07-19', '2024-07-23', 'Confirmed', 19),  
(20, 5, 20, '2024-07-20', '2024-07-24', 'Confirmed', 20)
```

```
mysql> SELECT * FROM Booking;
```

BookingID	GuestID	ListingID	CheckInDate	CheckOutDate	PaymentID	Status
1	1	1	2024-07-01	2024-07-05	1	Confirmed
2	2	2	2024-07-02	2024-07-06	2	Confirmed
3	3	3	2024-07-03	2024-07-07	3	Confirmed
4	4	4	2024-07-04	2024-07-08	4	Confirmed
5	5	5	2024-07-05	2024-07-09	5	Confirmed
6	1	6	2024-07-06	2024-07-10	6	Confirmed
7	2	7	2024-07-07	2024-07-11	7	Confirmed
8	3	8	2024-07-08	2024-07-12	8	Confirmed
9	4	9	2024-07-09	2024-07-13	9	Confirmed
10	5	10	2024-07-10	2024-07-14	10	Confirmed
11	1	11	2024-07-11	2024-07-15	11	Confirmed
12	2	12	2024-07-12	2024-07-16	12	Confirmed
13	3	13	2024-07-13	2024-07-17	13	Confirmed
14	4	14	2024-07-14	2024-07-18	14	Confirmed
15	5	15	2024-07-15	2024-07-19	15	Confirmed
16	1	16	2024-07-16	2024-07-20	16	Confirmed
17	2	17	2024-07-17	2024-07-21	17	Confirmed
18	3	18	2024-07-18	2024-07-22	18	Confirmed
19	4	19	2024-07-19	2024-07-23	19	Confirmed
20	5	20	2024-07-20	2024-07-24	20	Confirmed

20 rows in set (0.00 sec)

Country table implementation

Create Country table:

```
CREATE TABLE Country(  
  country_id INTEGER NOT NULL PRIMARY KEY,  
  COUNTRY_name character varying(100) NOT NULL,  
  country_code character varying(3) NOT NULL  
);
```

```
mysql> desc country;  
+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+  
| country_id | int | NO | PRI | NULL | |  
| COUNTRY_name | varchar(100) | NO | | NULL | |  
| country_code | varchar(3) | NO | | NULL | |  
+-----+-----+-----+-----+-----+  
3 rows in set (0.01 sec)
```

Insert Country values:

```
mysql> INSERT INTO Country VALUES  
(1, 'Netherlands', 'NL'),  
(2, 'Germany', 'DE'),  
(3, 'England', 'GB'),  
(4, 'Belgium', 'BE'),  
(5, 'Luxembourg', 'LU'),  
(6, 'France', 'FR'),  
(7, 'Spain', 'ES'),  
(8, 'Greece', 'GR'),  
(9, 'Italy', 'IT'),  
(10, 'Portugal', 'PT'),  
(11, 'United States', 'USA'),  
(12, 'Australia', 'AU'),  
(13, 'Denmark', 'DK'),  
(14, 'Poland', 'PL'),  
(15, 'Austria', 'AT'),  
(16, 'Netherland', 'NL'),  
(17, 'Norway', 'NO'),  
(18, 'Sweden', 'SE'),  
(19, 'Finland', 'FI'),  
(20, 'Latvia', 'LV'),  
(21, 'Lithuania', 'LT'),  
(22, 'Czechia', 'CZ'),  
(23, 'Mexico', 'MX'),  
(24, 'Brazil', 'BR'),  
(25, 'Argentina', 'AR'),  
(26, 'Chile', 'CL'),  
(27, 'Peru', 'PE'),  
(28, 'Japan', 'JP'),  
(29, 'South Korea', 'KR'),  
(30, 'China', 'CN'),  
(31, 'India', 'IN'),  
(32, 'Russia', 'RU'),  
(33, 'South Africa', 'ZA'),  
(34, 'Egypt', 'EG'),  
(35, 'Saudi Arabia', 'SA');
```

```
mysql> SELECT * FROM Country;  
+-----+-----+-----+  
| country_id | COUNTRY_name | country_code |  
+-----+-----+-----+  
| 1 | Netherlands | NL |  
| 2 | Germany | DE |  
| 3 | England | GB |  
| 4 | Belgium | BE |  
| 5 | Luxembourg | LU |  
| 6 | France | FR |  
| 7 | Spain | ES |  
| 8 | Greece | GR |  
| 9 | Italy | IT |  
| 10 | Portugal | PT |  
| 11 | United States | USA |  
| 12 | Australia | AU |  
| 13 | Denmark | DK |  
| 14 | Poland | PL |  
| 15 | Austria | AT |  
| 16 | Netherland | NL |  
| 17 | Norway | NO |  
| 18 | Sweden | SE |  
| 19 | Finland | FI |  
| 20 | Latvia | LV |  
| 21 | Lithuania | LT |  
| 22 | Czechia | CZ |  
| 23 | Mexico | MX |  
| 24 | Brazil | BR |  
| 25 | Argentina | AR |  
| 26 | Chile | CL |  
| 27 | Peru | PE |  
| 28 | Japan | JP |  
| 29 | South Korea | KR |  
| 30 | China | CN |  
| 31 | India | IN |  
| 32 | Russia | RU |  
| 33 | South Africa | ZA |  
| 34 | Egypt | EG |  
| 35 | Saudi Arabia | SA |
```

Create City table:

```
CREATE TABLE City (
    city_id INTEGER NOT NULL PRIMARY KEY,
    city_name VARCHAR(200),
    country_id INTEGER,
    FOREIGN KEY(country_id) REFERENCES Country(country_id)
);
```

```
mysql> desc city;
```

Field	Type	Null	Key	Default	Extra
city_id	int	NO	PRI	NULL	
city_name	varchar(200)	YES		NULL	
country_id	int	YES	MUL	NULL	

3 rows in set (0.00 sec)

Insert City values:

DESSERT CITY NAMES
 11, "Aachen", 11,
 12, "Aachen", 11,
 13, "The Hague", 11
 --- Germany
 DESSERT CITY NAMES
 14, "Berlin", 21,
 15, "Munich", 19,
 16, "Frankfurt", 21
 --- England
 DESSERT CITY NAMES
 17, "London", 19,
 18, "Manchester", 11,
 19, "Worthington", 11
 --- Belgium
 DESSERT CITY NAMES
 110, "Brussels", 41,
 111, "Antwerp", 41,
 112, "Ghent", 41
 --- Luxembourg
 DESSERT CITY NAMES
 113, "Luxembourg City", 1
 --- France
 DESSERT CITY NAMES
 114, "Paris", 11,
 115, "Marseille", 11,
 116, "Lyon", 11
 --- Spain
 DESSERT CITY NAMES
 117, "Madrid", 11,
 118, "Barcelona", 11,
 119, "Valencia", 11
 --- Greece
 DESSERT CITY NAMES
 120, "Athens", 11,
 121, "Thessalonika", 11

mysql SELECT * FROM City;		
city_id	city_name	country_id
1	Alexandria	1
2	Amsterdam	2
3	Atlanta	3
4	Berlin	4
5	Bombay	5
6	Frankfurt	2
7	Geneva	2
8	Hanover	3
9	Hong Kong	7
10	Los Angeles	3
11	Moscow	6
12	New York	3
13	Osaka	8
14	San Francisco	3
15	Shanghai	9
16	Singapore	10
17	Tokyo	8
18	Vienna	4
19	Washington	3
20	Zurich	2

Payment table implementation

Create Payment table:

```
CREATE TABLE Payment (  
  PaymentID INTEGER NOT NULL PRIMARY KEY,  
  Amount FLOAT,  
  PaymentDate DATE,  
  PaymentMethodID INTEGER,  
  FOREIGN KEY (PaymentMethodID) REFERENCES PaymentMethod (PaymentMethodID)  
);
```

```
mysql> desc payment;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| PaymentID | int | NO | PRI | NULL | |  
| Amount | float | YES | | NULL | |  
| PaymentDate | date | YES | | NULL | |  
| PaymentMethodID | int | YES | MUL | NULL | |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.00 sec)
```

Insert Payment table:

```
INSERT INTO Payment (PaymentID, Amount, PaymentDate, PaymentMethodID)  
VALUES  
(1, 100.50, '2024-07-04', 1),  
(2, 75.25, '2024-07-05', 2),  
(3, 150.00, '2024-07-06', 1),  
(4, 200.75, '2024-07-07', 3),  
(5, 50.00, '2024-07-08', 2),  
(6, 125.30, '2024-07-09', 4),  
(7, 180.90, '2024-07-10', 1),  
(8, 95.60, '2024-07-11', 2),  
(9, 120.25, '2024-07-12', 3),  
(10, 300.00, '2024-07-13', 1),  
(11, 85.75, '2024-07-14', 2),  
(12, 150.50, '2024-07-15', 4),  
(13, 210.80, '2024-07-16', 1),  
(14, 180.00, '2024-07-17', 1),  
(15, 95.25, '2024-07-18', 3),  
(16, 200.00, '2024-07-19', 1),  
(17, 120.75, '2024-07-20', 2),  
(18, 300.50, '2024-07-21', 4),  
(19, 150.25, '2024-07-22', 1),  
(20, 100.90, '2024-07-23', 2);
```

```
mysql> SELECT * FROM payment;  
+-----+-----+-----+-----+  
| PaymentID | Amount | PaymentDate | PaymentMethodID |  
+-----+-----+-----+-----+  
| 1 | 100.5 | 2024-07-04 | 1 |  
| 2 | 75.25 | 2024-07-05 | 2 |  
| 3 | 150 | 2024-07-06 | 1 |  
| 4 | 200.75 | 2024-07-07 | 3 |  
| 5 | 50 | 2024-07-08 | 2 |  
| 6 | 125.3 | 2024-07-09 | 4 |  
| 7 | 180.9 | 2024-07-10 | 1 |  
| 8 | 95.6 | 2024-07-11 | 2 |  
| 9 | 120.25 | 2024-07-12 | 3 |  
| 10 | 300 | 2024-07-13 | 1 |  
| 11 | 85.75 | 2024-07-14 | 2 |  
| 12 | 150.5 | 2024-07-15 | 4 |  
| 13 | 210.8 | 2024-07-16 | 1 |  
| 14 | 180 | 2024-07-17 | 1 |  
| 15 | 95.25 | 2024-07-18 | 3 |  
| 16 | 200 | 2024-07-19 | 1 |  
| 17 | 120.75 | 2024-07-20 | 2 |  
| 18 | 300.5 | 2024-07-21 | 4 |  
| 19 | 150.25 | 2024-07-22 | 1 |  
| 20 | 100.9 | 2024-07-23 | 2 |  
+-----+-----+-----+-----+  
20 rows in set (0.00 sec)  
  
mysql>
```

Complaint table implementation

Create Complaint table:

```
CREATE TABLE Complaint(  
  complaintID INTEGER NOT NULL PRIMARY KEY,  
  bookingID INTEGER,  
  complaint_text VARCHAR(200),  
  complaint_date DATE,  
  FOREIGN KEY(bookingID) REFERENCES Booking(bookingID)  
);
```

```
mysql> desc complaint;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| complaintID | int | NO | PRI | NULL | |  
| bookingID | int | YES | MUL | NULL | |  
| complaint_text | varchar(200) | YES | | NULL | |  
| complaint_date | date | YES | | NULL | |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.00 sec)
```

Insert Complaint values:

```
INSERT INTO Complaint (complaintID, bookingID, complaint_text, complaint_date)  
VALUES  
(1, 1, 'The apartment was not clean upon arrival.', '2024-01-15'),  
(2, 2, 'The host was unresponsive.', '2024-02-10'),  
(3, 3, 'There was no hot water in the shower.', '2024-03-05'),  
(4, 4, 'The Wi-Fi was not working.', '2024-03-22'),  
(5, 5, 'The property was not as described.', '2024-04-10'),  
(6, 6, 'The bed was uncomfortable.', '2024-04-25'),  
(7, 7, 'The host canceled the booking last minute.', '2024-05-05'),  
(8, 8, 'The location was noisy at night.', '2024-05-15'),  
(9, 9, 'The check-in process was very difficult.', '2024-05-25'),  
(10, 10, 'The amenities listed were not available.', '2024-06-01'),  
(11, 11, 'The heating system was not working.', '2024-06-01'),  
(12, 12, 'There were bugs in the apartment.', '2024-06-20'),  
(13, 13, 'The kitchen was not equipped as promised.', '2024-07-01'),  
(14, 14, 'The host was rude and unprofessional.', '2024-07-10'),  
(15, 15, 'The air conditioning was not working.', '2024-07-20'),  
(16, 16, 'There was no parking available as listed.', '2024-08-01'),  
(17, 17, 'The property had a bad smell.', '2024-08-15'),  
(18, 18, 'The host entered the property without notice.', '2024-08-25'),  
(19, 19, 'The apartment was too cold.', '2024-09-05'),  
(20, 20, 'The host charged extra fees not listed.', '2024-09-15');
```

```
20 rows in set (0.00 sec)  
  
mysql> SELECT * FROM Complaint;  
+-----+-----+-----+-----+  
| complaintID | bookingID | complaint_text | complaint_date |  
+-----+-----+-----+-----+  
| 1 | 1 | The apartment was not clean upon arrival. | 2024-01-15 |  
| 2 | 2 | The host was unresponsive. | 2024-02-10 |  
| 3 | 3 | There was no hot water in the shower. | 2024-03-05 |  
| 4 | 4 | The Wi-Fi was not working. | 2024-03-22 |  
| 5 | 5 | The property was not as described. | 2024-04-10 |  
| 6 | 6 | The bed was uncomfortable. | 2024-04-25 |  
| 7 | 7 | The host canceled the booking last minute. | 2024-05-05 |  
| 8 | 8 | The location was noisy at night. | 2024-05-15 |  
| 9 | 9 | The check-in process was very difficult. | 2024-05-25 |  
| 10 | 10 | The amenities listed were not available. | 2024-06-01 |  
| 11 | 11 | The heating system was not working. | 2024-06-01 |  
| 12 | 12 | There were bugs in the apartment. | 2024-06-20 |  
| 13 | 13 | The kitchen was not equipped as promised. | 2024-07-01 |  
| 14 | 14 | The host was rude and unprofessional. | 2024-07-10 |  
| 15 | 15 | The air conditioning was not working. | 2024-07-20 |  
| 16 | 16 | There was no parking available as listed. | 2024-08-01 |  
| 17 | 17 | The property had a bad smell. | 2024-08-15 |  
| 18 | 18 | The host entered the property without notice. | 2024-08-25 |  
| 19 | 19 | The apartment was too cold. | 2024-09-05 |  
| 20 | 20 | The host charged extra fees not listed. | 2024-09-15 |  
+-----+-----+-----+-----+  
20 rows in set (0.00 sec)
```


Message table implementation

Create Message table:

```
CREATE TABLE Message (
  messageId INTEGER NOT NULL PRIMARY KEY,
  senderID INTEGER,
  receiverID INTEGER,
  content VARCHAR(200),
  CONSTRAINT sender_fk FOREIGN KEY (senderID) REFERENCES sender (senderID),
  CONSTRAINT receiver_fk FOREIGN KEY (receiverID) REFERENCES receiver (receiverID)
);
```

```
mysql> desc Message;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| messageId | int | NO | PRI | NULL | |
| senderID | int | YES | MUL | NULL | |
| receiverID | int | YES | MUL | NULL | |
| content | varchar(200) | YES | | NULL | |
+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

Insert Message values:

```
INSERT INTO Message (messageID, senderID, receiverID, content)
VALUES
(1, 181, 182, 'Hey there, how are you?'),
(2, 182, 181, 'Hi! I\'m doing well, thanks! How about you?'),
(3, 181, 182, 'Hi Charlie, are we still meeting tomorrow?'),
(4, 182, 181, 'Yes, at 3 PM. See you then!'),
(5, 184, 185, 'Hey Eve, are you free for coffee next week?'),
(6, 185, 184, 'Sure! I\'m available on Thursday.'),
(7, 181, 185, 'Alice, have you finalized the presentation slides?'),
(8, 185, 181, 'Yes, I\'ll send them over by tomorrow morning.'),
(9, 182, 183, 'Charlie, do you need help with the project update?'),
(10, 183, 182, 'Yes, could you review the latest draft?'),
(11, 184, 181, 'Alice, when is the deadline for the report?'),
(12, 181, 184, 'It\'s due by the end of the week.'),
(13, 182, 185, 'Eve, let\'s discuss the marketing strategy this afternoon.'),
(14, 185, 182, 'Sure! I\'ll be available after lunch.'),
(15, 183, 184, 'Dana, can you attend the meeting tomorrow?'),
(16, 184, 183, 'I have another commitment, but I\'ll join remotely.'),
(17, 185, 183, 'Charlie, do you have the latest financial report?'),
(18, 183, 185, 'Yes, I\'ll share it with you shortly.'),
(19, 181, 184, 'Alice, can we schedule a call to discuss the project status?'),
(20, 184, 181, 'Sure! How about tomorrow morning?');
```

```
mysql> SELECT * FROM Message;
+-----+-----+-----+-----+
| messageId | senderID | receiverID | content |
+-----+-----+-----+-----+
| 1 | 181 | 182 | Hey there, how are you? |
| 2 | 182 | 181 | Hi! I'm doing well, thanks! How about you? |
| 3 | 181 | 182 | Hi Charlie, are we still meeting tomorrow? |
| 4 | 182 | 181 | Yes, at 3 PM. See you then! |
| 5 | 184 | 185 | Hey Eve, are you free for coffee next week? |
| 6 | 185 | 184 | Sure! I'm available on Thursday. |
| 7 | 181 | 185 | Alice, have you finalized the presentation slides? |
| 8 | 185 | 181 | Yes, I'll send them over by tomorrow morning. |
| 9 | 182 | 183 | Charlie, do you need help with the project update? |
| 10 | 183 | 182 | Yes, could you review the latest draft? |
| 11 | 184 | 181 | Alice, when is the deadline for the report? |
| 12 | 181 | 184 | It's due by the end of the week. |
| 13 | 182 | 185 | Eve, let's discuss the marketing strategy this afternoon. |
| 14 | 185 | 182 | Sure! I'll be available after lunch. |
| 15 | 183 | 184 | Dana, can you attend the meeting tomorrow? |
| 16 | 184 | 183 | I have another commitment, but I'll join remotely. |
| 17 | 185 | 183 | Charlie, do you have the latest financial report? |
| 18 | 183 | 185 | Yes, I'll share it with you shortly. |
| 19 | 181 | 184 | Alice, can we schedule a call to discuss the project status? |
| 20 | 184 | 181 | Sure! How about tomorrow morning? |
+-----+-----+-----+-----+
```

Report table implementation

Create Report table:

```
CREATE TABLE Report (
  ReportID INT(8) NOT NULL PRIMARY KEY,
  ReporterID INT(4),
  ListingID INT(4),
  ReportReason VARCHAR(255),
  ReportDate DATE,
  ReportStatus VARCHAR(20),
  FOREIGN KEY (ReporterID) REFERENCES User(userID),
  FOREIGN KEY (ListingID) REFERENCES Listing(ListingID);
);
```

```
mysql> desc report;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ReportID | int | NO | PRI | NULL |  |
| ReporterID | int | YES | MLI | NULL |  |
| ListingID | int | YES | MLI | NULL |  |
| ReportReason | varchar(255) | YES |  | NULL |  |
| ReportDate | date | YES |  | NULL |  |
| ReportStatus | varchar(20) | YES |  | NULL |  |
+-----+-----+-----+-----+-----+-----+
0 rows in set (0.01 sec)
```

Insert Report table:

```
mysql> INSERT INTO Report (ReporterID, ListingID, ReportReason, ReportDate, ReportStatus)
VALUES
(1, 1, 'Noise disturbance', '2024-07-01', 'Pending'),
(2, 2, 'Unclean property', '2024-07-02', 'Reviewed'),
(3, 3, 'Unauthorized pets', '2024-07-03', 'Resolved'),
(4, 4, 'Broken amenities', '2024-07-04', 'Pending'),
(5, 5, 'Inaccurate listing', '2024-07-05', 'Pending'),
(6, 1, 6, 'Unresponsive host', '2024-07-06', 'Reviewed'),
(7, 2, 7, 'Unauthorized guests', '2024-07-07', 'Resolved'),
(8, 3, 8, 'Property damage', '2024-07-08', 'Pending'),
(9, 4, 9, 'Late check-in', '2024-07-09', 'Reviewed'),
(10, 5, 10, 'Inaccurate photos', '2024-07-10', 'Pending'),
(11, 1, 11, 'Unclean linens', '2024-07-11', 'Reviewed'),
(12, 2, 12, 'Noisy neighbors', '2024-07-12', 'Resolved'),
(13, 3, 13, 'Faulty locks', '2024-07-13', 'Pending'),
(14, 4, 14, 'Overbooked property', '2024-07-14', 'Reviewed'),
(15, 5, 15, 'Host canceled booking', '2024-07-15', 'Pending'),
(16, 1, 16, 'Water leakage', '2024-07-16', 'Reviewed'),
(17, 2, 17, 'Unsanitary conditions', '2024-07-17', 'Resolved'),
(18, 3, 18, 'Misleading reviews', '2024-07-18', 'Pending'),
(19, 4, 19, 'Poor customer service', '2024-07-19', 'Reviewed'),
(20, 5, 20, 'Fraudulent activity', '2024-07-20', 'Resolved');
```

```
mysql> select * from report;
+-----+-----+-----+-----+-----+-----+
| ReportID | ReporterID | ListingID | ReportReason | ReportDate | ReportStatus |
+-----+-----+-----+-----+-----+-----+
| 1 | 1 | 1 | Noise disturbance | 2024-07-01 | Pending |
| 2 | 2 | 2 | Unclean property | 2024-07-02 | Reviewed |
| 3 | 3 | 3 | Unauthorized pets | 2024-07-03 | Resolved |
| 4 | 4 | 4 | Broken amenities | 2024-07-04 | Pending |
| 5 | 5 | 5 | Inaccurate listing | 2024-07-05 | Pending |
| 6 | 1 | 6 | Unresponsive host | 2024-07-06 | Reviewed |
| 7 | 2 | 7 | Unauthorized guests | 2024-07-07 | Resolved |
| 8 | 3 | 8 | Property damage | 2024-07-08 | Pending |
| 9 | 4 | 9 | Late check-in | 2024-07-09 | Reviewed |
| 10 | 5 | 10 | Inaccurate photos | 2024-07-10 | Pending |
| 11 | 1 | 11 | Unclean linens | 2024-07-11 | Reviewed |
| 12 | 2 | 12 | Noisy neighbors | 2024-07-12 | Resolved |
| 13 | 3 | 13 | Faulty locks | 2024-07-13 | Pending |
| 14 | 4 | 14 | Overbooked property | 2024-07-14 | Reviewed |
| 15 | 5 | 15 | Host canceled booking | 2024-07-15 | Pending |
| 16 | 1 | 16 | Water leakage | 2024-07-16 | Reviewed |
| 17 | 2 | 17 | Unsanitary conditions | 2024-07-17 | Resolved |
| 18 | 3 | 18 | Misleading reviews | 2024-07-18 | Pending |
| 19 | 4 | 19 | Poor customer service | 2024-07-19 | Reviewed |
| 20 | 5 | 20 | Fraudulent activity | 2024-07-20 | Resolved |
+-----+-----+-----+-----+-----+-----+
20 rows in set (0.01 sec)
```

Review table implementation

Create Review table:

```
CREATE TABLE Review(
  ReviewID INTEGER NOT NULL PRIMARY KEY,
  ReviewerID INTEGER,
  Rating FLOAT,
  Comments VARCHAR(200),
  ReviewDate DATE,
  FOREIGN KEY(ReviewerID) REFERENCES User(userID)
);
```

```
mysql> desc review;
```

Field	Type	Null	Key	Default	Extra
ReviewID	int	NO	PRI	NULL	
ReviewerID	int	YES	MUL	NULL	
Rating	float	YES		NULL	
Comments	varchar(200)	YES		NULL	
ReviewDate	date	YES		NULL	

5 rows in set (0.01 sec)

Insert Review values:

```
INSERT INTO Review (ReviewID, ReviewerID, Rating, Comments, ReviewDate)
VALUES
(1, 1, 4.5, 'Great experience, highly recommended!', '2024-01-15'),
(2, 2, 3.0, 'Average service, room for improvement.', '2024-02-10'),
(3, 3, 5.0, 'Excellent! Will definitely come back.', '2024-03-05'),
(4, 4, 2.5, 'Not as expected, needs better maintenance.', '2024-04-12'),
(5, 5, 4.0, 'Very good stay, friendly staff.', '2024-05-20'),
(6, 6, 3.5, 'Good value for the price.', '2024-06-15'),
(7, 7, 4.8, 'Fantastic place, very clean and comfortable.', '2024-07-01'),
(8, 8, 2.0, 'Poor service, will not recommend.', '2024-07-10'),
(9, 9, 4.2, 'Nice location, enjoyed the stay.', '2024-07-18'),
(10, 10, 3.8, 'Decent place, could be better.', '2024-08-05'),
(11, 11, 4.7, 'Loved it! Beautiful and serene.', '2024-08-22'),
(12, 12, 3.3, 'Okay, but some issues with cleanliness.', '2024-09-10'),
(13, 13, 5.0, 'Perfect stay, everything was just right.', '2024-09-25'),
(14, 14, 2.8, 'Below average, not worth the price.', '2024-10-12'),
(15, 15, 4.6, 'Great service, very accommodating.', '2024-10-30'),
(16, 16, 3.2, 'Good experience, but could improve.', '2024-11-10'),
(17, 17, 4.9, 'Exceptional! Highly recommend.', '2024-11-25'),
(18, 18, 2.2, 'Not satisfied, needs major improvements.', '2024-12-05'),
(19, 19, 4.1, 'Nice place, enjoyed the stay.', '2024-12-15'),
(20, 20, 3.9, 'Overall good, but some minor issues.', '2024-12-30');
```

```
mysql> SELECT * FROM review;
```

ReviewID	ReviewerID	Rating	Comments	ReviewDate
1	1	4.5	Great experience, highly recommended!	2024-01-15
2	2	3	Average service, room for improvement.	2024-02-10
3	3	5	Excellent! Will definitely come back.	2024-03-05
4	4	2.5	Not as expected, needs better maintenance.	2024-04-12
5	5	4	Very good stay, friendly staff.	2024-05-20
6	6	3.5	Good value for the price.	2024-06-15
7	7	4.8	Fantastic place, very clean and comfortable.	2024-07-01
8	8	2	Poor service, will not recommend.	2024-07-10
9	9	4.2	Nice location, enjoyed the stay.	2024-07-18
10	10	3.8	Decent place, could be better.	2024-08-05
11	11	4.7	Loved it! Beautiful and serene.	2024-08-22
12	12	3.3	Okay, but some issues with cleanliness.	2024-09-10
13	13	5	Perfect stay, everything was just right.	2024-09-25
14	14	2.8	Below average, not worth the price.	2024-10-12
15	15	4.6	Great service, very accommodating.	2024-10-30
16	16	3.7	Good experience, but could improve.	2024-11-10
17	17	4.9	Exceptional! Highly recommend.	2024-11-25
18	18	2.2	Not satisfied, needs major improvements.	2024-12-05
19	19	4.1	Nice place, enjoyed the stay.	2024-12-15
20	20	3.9	Overall good, but some minor issues.	2024-12-30

20 rows in set (0.00 sec)

Tax table implementation

Create Tax table:

```
CREATE TABLE Tax(  
  TaxID INTEGER NOT NULL PRIMARY KEY,  
  BookingID INTEGER,  
  Amount FLOAT,  
  FOREIGN KEY(BookingID) REFERENCES Booking(BookingID)  
);
```

```
mysql> desc tax;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| TaxID | int  | NO   | PRI | NULL    |       |  
| BookingID | int | YES  | MUL | NULL    |       |  
| Amount | float | YES  |     |         |       |  
+-----+-----+-----+-----+-----+-----+  
3 rows in set (0.00 sec)
```

Insert Tax values:

```
INSERT INTO Tax (TaxID, BookingID, Amount) VALUES  
(1, 1, 10.5),  
(2, 2, 12.0),  
(3, 3, 8.7),  
(4, 4, 15.2),  
(5, 5, 9.8),  
(6, 6, 11.3),  
(7, 7, 14.5),  
(8, 8, 7.2),  
(9, 9, 16.0),  
(10, 10, 10.9),  
(11, 11, 13.4),  
(12, 12, 11.1),  
(13, 13, 9.5),  
(14, 14, 12.8),  
(15, 15, 8.3),  
(16, 16, 14.9),  
(17, 17, 6.7),  
(18, 18, 13.7),  
(19, 19, 10.2),  
(20, 20, 12.5);
```

```
mysql> SELECT * FROM tax;  
+-----+-----+-----+  
| TaxID | BookingID | Amount |  
+-----+-----+-----+  
| 1     | 1         | 10.5   |  
| 2     | 2         | 12.0   |  
| 3     | 3         | 8.7    |  
| 4     | 4         | 15.2   |  
| 5     | 5         | 9.8    |  
| 6     | 6         | 11.3   |  
| 7     | 7         | 14.5   |  
| 8     | 8         | 7.2    |  
| 9     | 9         | 16.0   |  
| 10    | 10        | 10.9   |  
| 11    | 11        | 13.4   |  
| 12    | 12        | 11.1   |  
| 13    | 13        | 9.5    |  
| 14    | 14        | 12.8   |  
| 15    | 15        | 8.3    |  
| 16    | 16        | 14.9   |  
| 17    | 17        | 6.7    |  
| 18    | 18        | 13.7   |  
| 19    | 19        | 10.2   |  
| 20    | 20        | 12.5   |  
+-----+-----+-----+  
20 rows in set (0.01 sec)
```

Transaction table implementation

Create Transaction table:

```
CREATE TABLE Transaction (
  TransactionID INTEGER NOT NULL PRIMARY KEY,
  UserID INTEGER,
  BookingID INTEGER,
  Amount FLOAT,
  FOREIGN KEY (UserID) REFERENCES User(UserID),
  FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)
);
```

mysql> desc transaction;

Field	Type	Null	Key	Default	Extra
TransactionID	int	NO	PRI	NULL	
UserID	int	YES	MUL	NULL	
BookingID	int	YES	MUL	NULL	
Amount	float	YES		NULL	

rows in set (0.00 sec)

Insert Transaction values:

```
INSERT INTO Transaction (TransactionID, UserID, BookingID, Amount)
VALUES
(1, 1, 1, 100.00),
(2, 2, 2, 250.00),
(3, 3, 3, 80.00),
(4, 4, 4, 150.00),
(5, 5, 5, 120.00),
(6, 1, 6, 200.00),
(7, 2, 7, 500.00),
(8, 3, 8, 180.00),
(9, 4, 9, 250.00),
(10, 5, 10, 120.00),
(11, 1, 11, 300.00),
(12, 2, 12, 400.00),
(13, 3, 13, 150.00),
(14, 4, 14, 600.00),
(15, 5, 15, 800.00),
(16, 1, 16, 80.00),
(17, 2, 17, 120.00),
(18, 3, 18, 100.00),
(19, 4, 19, 350.00),
(20, 5, 20, 150.00);
```

mysql> SELECT * FROM transaction;

TransactionID	UserID	BookingID	Amount
1	1	1	100
2	2	2	250
3	3	3	80
4	4	4	150
5	5	5	120
6	1	6	200
7	2	7	500
8	3	8	180
9	4	9	250
10	5	10	120
11	1	11	300
12	2	12	400
13	3	13	150
14	4	14	600
15	5	15	800
16	1	16	80
17	2	17	120
18	3	18	100
19	4	19	350
20	5	20	150

20 rows in set (0.01 sec)

Social media account implementation

Create social media account table:

```
CREATE TABLE Transaction (
  TransactionID INTEGER NOT NULL PRIMARY KEY,
  UserID INTEGER,
  BookingID INTEGER,
  Amount FLOAT,
  FOREIGN KEY (UserID) REFERENCES User(UserID),
  FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)
);
```

```
mysql> desc transaction;
```

Field	Type	Null	Key	Default	Extra
TransactionID	int	NO	PRI	NULL	
UserID	int	YES	MUL	NULL	
BookingID	int	YES	MUL	NULL	
Amount	float	YES		NULL	

```
rows in set (0.00 sec)
```

Insert social media accounts values:

```
INSERT INTO UserSocialMedia (user_social_media_id, user_id, platform_id, profile_url, username) VALUES
(1, 1, 1, 'https://facebook.com/alice.smith', 'alice.smith'),
(2, 1, 2, 'https://twitter.com/alice.smith', 'alice.smith'),
(3, 2, 1, 'https://facebook.com/bob.johnson', 'bob.johnson'),
(4, 2, 3, 'https://linkedin.com/in/bobjohnson', 'bobjohnson'),
(5, 3, 4, 'https://instagram.com/charlie.brown', 'charlie.brown'),
(6, 3, 2, 'https://twitter.com/charlie.brown', 'charlie.brown'),
(7, 4, 3, 'https://linkedin.com/in/danawhite', 'danawhite'),
(8, 4, 1, 'https://facebook.com/dana.white', 'dana.white'),
(9, 5, 2, 'https://twitter.com/eve.black', 'eve.black'),
(10, 5, 4, 'https://instagram.com/eve.black', 'eve.black'),
(11, 6, 3, 'https://linkedin.com/in/frankadams', 'frankadams'),
(12, 6, 1, 'https://facebook.com/frank.adams', 'frank.adams'),
(13, 7, 4, 'https://instagram.com/grace.taylor', 'grace.taylor'),
(14, 7, 3, 'https://linkedin.com/in/gracetaylor', 'gracetaylor'),
(15, 8, 1, 'https://facebook.com/henry.lee', 'henry.lee'),
(16, 8, 2, 'https://twitter.com/henry.lee', 'henry.lee'),
(17, 9, 4, 'https://instagram.com/ivy.clark', 'ivy.clark'),
(18, 9, 3, 'https://linkedin.com/in/ivyclark', 'ivyclark'),
(19, 10, 1, 'https://facebook.com/jack.harris', 'jack.harris'),
(20, 10, 2, 'https://twitter.com/jack.harris', 'jack.harris');
```

```
mysql> SELECT * FROM UserSocialMedia;
mysql> SELECT * FROM UserSocialMedia;
```

user_social_media_id	user_id	platform_id	profile_url	username
1	1	1	https://facebook.com/alice.smith	alice.smith
2	1	2	https://twitter.com/alice.smith	alice.smith
3	2	1	https://facebook.com/bob.johnson	bob.johnson
4	2	3	https://linkedin.com/in/bobjohnson	bobjohnson
5	3	4	https://instagram.com/charlie.brown	charlie.brown
6	3	2	https://twitter.com/charlie.brown	charlie.brown
7	4	3	https://linkedin.com/in/danawhite	danawhite
8	4	1	https://facebook.com/dana.white	dana.white
9	5	2	https://twitter.com/eve.black	eve.black
10	5	4	https://instagram.com/eve.black	eve.black
11	6	3	https://linkedin.com/in/frankadams	frankadams
12	6	1	https://facebook.com/frank.adams	frank.adams
13	7	4	https://instagram.com/grace.taylor	grace.taylor
14	7	3	https://linkedin.com/in/gracetaylor	gracetaylor
15	8	1	https://facebook.com/henry.lee	henry.lee
16	8	2	https://twitter.com/henry.lee	henry.lee
17	9	4	https://instagram.com/ivy.clark	ivy.clark
18	9	3	https://linkedin.com/in/ivyclark	ivyclark
19	10	1	https://facebook.com/jack.harris	jack.harris
20	10	2	https://twitter.com/jack.harris	jack.harris

```
20 rows in set (0.00 sec)
```


Notification table implementation

Create Notification table:

```
CREATE TABLE Notification(  
  notificationID INTEGER NOT NULL PRIMARY KEY,  
  senderID INTEGER,  
  receiverID INTEGER,  
  textMessage VARCHAR(200),  
  notification_date DATE,  
  FOREIGN KEY(senderID) REFERENCES sender(userID),  
  FOREIGN KEY (receiverID) REFERENCES receiver(userID)  
);
```

```
mysql> desc notification;
```

Field	Type	Null	Key	Default	Extra
notificationID	int	NO	PRI	NULL	
senderID	int	YES	MUL	NULL	
receiverID	int	YES	MUL	NULL	
textMessage	varchar(200)	YES		NULL	
notification_date	date	YES		NULL	

5 rows in set (0.00 sec)

Create Notification values:

```
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (1, 1, 2, 'Hello, this is a test message.', '2023-01-01');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (2, 2, 3, 'Your appointment is confirmed.', '2023-01-02');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (3, 3, 4, 'Please update your profile.', '2023-01-03');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (4, 4, 5, 'New message received.', '2023-01-04');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (5, 5, 6, 'Your order has been shipped.', '2023-01-05');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (6, 6, 7, 'Meeting scheduled for tomorrow.', '2023-01-06');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (7, 7, 8, 'Please review the document.', '2023-01-07');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (8, 8, 9, 'Your subscription has been renewed.', '2023-01-08');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (9, 9, 10, 'Password reset request.', '2023-01-09');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (10, 10, 11, 'Your account has been updated.', '2023-01-10');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (11, 11, 12, 'New friend request.', '2023-01-11');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (12, 12, 13, 'Please verify your email.', '2023-01-12');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (13, 13, 14, 'Event invitation received.', '2023-01-13');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (14, 14, 15, 'New comment on your post.', '2023-01-14');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (15, 15, 16, 'System maintenance scheduled.', '2023-01-15');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (16, 16, 17, 'Your ticket has been closed.', '2023-01-16');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (17, 17, 18, 'Congratulations on your achievement!', '2023-01-17');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (18, 18, 19, 'Your feedback is appreciated.', '2023-01-18');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (19, 19, 20, 'New update available.', '2023-01-19');  
mysql> INSERT INTO Notification (notificationID, senderID, receiverID, textMessage, notification_date) VALUES (20, 20, 1, 'Your request has been approved.', '2023-01-20');
```

```
mysql> SELECT * FROM Notification;
```

notificationID	senderID	receiverID	textMessage	notification_date
1	1	2	Hello, this is a test message.	2023-01-01
2	2	3	Your appointment is confirmed.	2023-01-02
3	3	4	Please update your profile.	2023-01-03
4	4	5	New message received.	2023-01-04
5	5	6	Your order has been shipped.	2023-01-05
6	6	7	Meeting scheduled for tomorrow.	2023-01-06
7	7	8	Please review the document.	2023-01-07
8	8	9	Your subscription has been renewed.	2023-01-08
9	9	10	Password reset request.	2023-01-09
10	10	11	Your account has been updated.	2023-01-10
11	11	12	New friend request.	2023-01-11
12	12	13	Please verify your email.	2023-01-12
13	13	14	Event invitation received.	2023-01-13
14	14	15	New comment on your post.	2023-01-14
15	15	16	System maintenance scheduled.	2023-01-15
16	16	17	Your ticket has been closed.	2023-01-16
17	17	18	Congratulations on your achievement!	2023-01-17
18	18	19	Your feedback is appreciated.	2023-01-18
19	19	20	New update available.	2023-01-19
20	20	1	Your request has been approved.	2023-01-20

20 rows in set (0.00 sec)

```
mysql>
```

Ln 2

Image table implementation

Create Image table:

```
CREATE TABLE Image(  
  imageID INTEGER NOT NULL PRIMARY KEY,  
  image_url VARCHAR(200),  
  mediaType VARCHAR(100)  
);
```

```
mysql> desc image;
```

Field	Type	Null	Key	Default	Extra
imageID	int	NO	PRI	NULL	
image_url	varchar(200)	YES		NULL	
mediaType	varchar(100)	YES		NULL	

Insert Image values:

```
mentation > development > insert_values > image.sql  
INSERT INTO Image (imageID, image_url, mediaType)  
VALUES  
(1, 'https://example.com/images/image1.jpg', 'JPEG'),  
(2, 'https://example.com/images/image2.png', 'PNG'),  
(3, 'https://example.com/images/image3.gif', 'GIF'),  
(4, 'https://example.com/images/image4.bmp', 'BMP'),  
(5, 'https://example.com/images/image5.webp', 'WEBP'),  
(6, 'https://example.com/images/image6.tiff', 'TIFF'),  
(7, 'https://example.com/images/image7.svg', 'SVG'),  
(8, 'https://example.com/images/image8.jpg', 'JPEG'),  
(9, 'https://example.com/images/image9.png', 'PNG'),  
(10, 'https://example.com/images/image10.gif', 'GIF'),  
(11, 'https://example.com/images/image11.bmp', 'BMP'),  
(12, 'https://example.com/images/image12.webp', 'WEBP'),  
(13, 'https://example.com/images/image13.tiff', 'TIFF'),  
(14, 'https://example.com/images/image14.svg', 'SVG'),  
(15, 'https://example.com/images/image15.jpg', 'JPEG'),  
(16, 'https://example.com/images/image16.png', 'PNG'),  
(17, 'https://example.com/images/image17.gif', 'GIF'),  
(18, 'https://example.com/images/image18.bmp', 'BMP'),  
(19, 'https://example.com/images/image19.webp', 'WEBP'),  
(20, 'https://example.com/images/image20.tiff', 'TIFF');
```

```
mysql> SELECT * FROM Image;
```

imageID	image_url	mediaType
1	https://example.com/images/image1.jpg	JPEG
2	https://example.com/images/image2.png	PNG
3	https://example.com/images/image3.gif	GIF
4	https://example.com/images/image4.bmp	BMP
5	https://example.com/images/image5.webp	WEBP
6	https://example.com/images/image6.tiff	TIFF
7	https://example.com/images/image7.svg	SVG
8	https://example.com/images/image8.jpg	JPEG
9	https://example.com/images/image9.png	PNG
10	https://example.com/images/image10.gif	GIF
11	https://example.com/images/image11.bmp	BMP
12	https://example.com/images/image12.webp	WEBP
13	https://example.com/images/image13.tiff	TIFF
14	https://example.com/images/image14.svg	SVG
15	https://example.com/images/image15.jpg	JPEG
16	https://example.com/images/image16.png	PNG
17	https://example.com/images/image17.gif	GIF
18	https://example.com/images/image18.bmp	BMP
19	https://example.com/images/image19.webp	WEBP
20	https://example.com/images/image20.tiff	TIFF

20 rows in set (0.00 sec)

Amenity table implementation

Create Amenity table:

```
CREATE TABLE Amenity(  
  amenityID INTEGER NOT NULL PRIMARY KEY,  
  amenityName VARCHAR(200)  
)
```

```
mysql> desc amenity;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| amenityID | int | NO | PRI | NULL | |  
| amenityName | varchar(200) | YES | | NULL | |  
+-----+-----+-----+-----+-----+-----+  
2 rows in set (0.00 sec)
```

Insert Amenity values:

```
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (1, 'Wifi');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (2, 'Air Conditioning');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (3, 'Heating');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (4, 'Kitchen');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (5, 'Washer');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (6, 'Dryer');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (7, 'Free Parking');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (8, 'Gym');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (9, 'Pool');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (10, 'Hot Tub');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (11, 'BBQ Grill');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (12, 'Fireplace');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (13, 'Breakfast');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (14, 'Workspace');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (15, 'TV');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (16, 'Hair Dryer');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (17, 'Iron');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (18, 'Laptop-friendly workspace');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (19, 'Shampoo');  
mysql> INSERT INTO Amenity (amenityID, amenityName) VALUES (20, 'Essentials');
```

```
mysql> SELECT * FROM Amenity;  
+-----+-----+  
| amenityID | amenityName |  
+-----+-----+  
| 1 | Wifi |  
| 2 | Air Conditioning |  
| 3 | Heating |  
| 4 | Kitchen |  
| 5 | Washer |  
| 6 | Dryer |  
| 7 | Free Parking |  
| 8 | Gym |  
| 9 | Pool |  
| 10 | Hot Tub |  
| 11 | BBQ Grill |  
| 12 | Fireplace |  
| 13 | Breakfast |  
| 14 | Workspace |  
| 15 | TV |  
| 16 | Hair Dryer |  
| 17 | Iron |  
| 18 | Laptop-friendly workspace |  
| 19 | Shampoo |  
| 20 | Essentials |  
+-----+-----+  
20 rows in set (0.00 sec)
```

Payment method table implementation

Create Payment method table:

```
CREATE TABLE PaymentMethod(  
  PaymentMethodID INTEGER NOT NULL PRIMARY KEY,  
  MethodName VARCHAR(200)  
);
```

```
mysql> desc paymentMethod;
```

Field	Type	Null	Key	Default	Extra
PaymentMethodID	int	NO	PRI	NULL	
MethodName	varchar(200)	YES		NULL	

2 rows in set (0.00 sec)

Insert payment methods values:

```
INSERT INTO PaymentMethod (PaymentMethodID, MethodName)  
VALUES  
(1, 'Credit Card'),  
(2, 'Debit Card'),  
(3, 'PayPal'),  
(4, 'Bank Transfer');
```

```
mysql> SELECT * FROM PaymentMethod;
```

PaymentMethodID	MethodName
1	Credit Card
2	Debit Card
3	PayPal
4	Bank Transfer

4 rows in set (0.01 sec)

Test cases

Test case: In this test we will try to query data from database using a programming language.

```
1 import unittest
2 import pymysql
3
4 class TestSQLQueries(unittest.TestCase):
5
6     def test_select_all_employees(self):
7         conn = pymysql.connect(host='localhost', user='root', password='samidsamid', db='airbnb')
8         cursor = conn.cursor()
9         cursor.execute("SELECT * FROM host")
10        result = cursor.fetchall()
11        expected = [
12            (1, 1, 4.5),
13            (2, 2, 4.0),
14            (3, 3, 3.5),
15            (4, 4, 4.2),
16            (5, 5, 4.0),
17            (6, 6, 4.7),
18            (7, 7, 4.3),
19            (8, 8, 4.6),
20            (9, 9, 4.1),
21            (10, 10, 3.9),
22            (11, 11, 4.4),
23            (12, 12, 4.9),
24            (13, 13, 3.0),
25            (14, 14, 4.0),
26            (15, 15, 4.3),
27            (16, 16, 4.2),
28            (17, 17, 4.7),
29            (18, 18, 3.6),
30            (19, 19, 4.3),
31            (20, 20, 4.8)
32        ]
33
34        self.assertEqual(result, expected)
```

Python script that fetch data
from database

```
*
-----
Ran 1 test in 0.004s
OK
python3 test_sql_queries.py
```

Test is accepted



Thank you!!

