Development of a Data Mart using MySQL

In order for organizations to store important data regarding clients, staffs, and services in an organized manner, various means of storing data can be employed such as Comma Separated Values (CSV's), Excel Spreadsheets, databases etc. However, only one will be used for this project which is the database. Database is a great way of storing and organizing data because it stores the data in an efficient manner, access management and functionality to databases is made possible through a database management system (DBMS). MySQL is a type of Relational database management which will be used to store and organize data in a relational manner. The aim of this project is to create a sample database about customers and services of a rental company for holiday destination such as Airbnb.

I am planning to build a data mart that can be used to infer data about clients, browse information about existing reservation, closed reservation, reservation history, customers review and so on. I will start this project by downloading MySQL and MySQL workbench to create the database after completion of schema design through Entity Relationship (ER) diagram.

After surfing Airbnb website for important details about the rental system works, I designed an Entity Relationship model that will be used as a background for designing this database, Entity relationship diagrams are used to display data schemas in a database. ER diagrams contains keys, relationships, entities, and attributes, three popular methods used for ER diagrams are: -

- Chen notation,
- Martin notation, and
- UML class diagram

However, UML class diagram is chosen to model this database schema, it can be observed in the UML diagram below that there are relationships between twenty tables in the UML diagram below, these tables are used to store various information about the reservation system, staffs, customers, available destinations etc. In order to maintain referential integrity, I will make sure that there is an actual primary key for every foreign key when using the designed UML diagram to create the database.

In the UML diagram below: - The **Customer** table will serve as the main table in which other tables relate to, this table will contain data about background information about customers such as name, age, gender, job title, foreign keys referencing other tables etc.

The **Dependendent** table will contain data about dependents of the customer which will be included in the reservation such as spouse, children etc.

The **Staff** table will contain information about which staff is in charge of the comfortability of the customer during his reservation such as staff name, rank and foreign key to the post inspection table.

The **Post Inspection** table will contain information about state of facilities and amenities after the guest has checked out of their reservation which will be carried out by the staff.

The **Address** table will contain information about the customers home address, postcode, street etc.

The **Staff, Post Inspection, Entertainment after use, Amenities after use, Facilities after use** tables will contain information about staffs, the customers they were assigned to and data

about details about their inspection of facilities, amenities after customers assigned to them has checked out.

The **Country** table will contain information about the country in which the customers reside in, it will also contain information about the country in which the room owners reside in. This table has two relationships attached to it.

The **Log-in** table will contain data about the customers login information such as email, phone number, account status and foreign key to the login details table.

The **Login Details** table will contain information about username, password, last updated date and foreign key to the security details table.

The **Security Questions** table will contain information about customers security questions in case of a situation in which a customer loses their login details.

The **Reservation** table will contain information about available locations such as, reservation type, booking date, start date etc. It will also foreign keys to maintain relationships with room and payment tables.

The **Room** table will contain information about various available room types and other information such as number of bedrooms, number of bathrooms, publish date etc. This table will also contain foreign keys relating to other tables namely room address and rules.

The **Room Address** table will contain information about the address the rooms are located at with street, state and country names. This table will also contain foreign keys relating to room owner and room amenities tables.

The **Room Owner** table will contain information about the owner of the published reservation space such as name, address etc.

The **Rules** table will contain information about dos and don’ts of each available room such as permission of pets and smoking on the property.

The **Payment** table will contain information about the room prices, discounts (if applicable), payment mode, payment status etc. This table will also contain a foreign key related to the reviews table.

The **Amenities** table will contain information about amenities on each property such as kitchen, air conditioning, heater, internet and so on. This table will also contain a foreign key related to the facilities table.

The **Facilities** table will contain information about facilities available on each available space such as hot tub, pool, gym, free parking etc. It will also contain a foreign key to the entertainment table.

The **Entertainment** table will contain information about the availability of entertainment equipment’s such as Televisions, Sound systems, beach front, water front’s etc.

In the process of normalizing the database during creation using MySQL, more tables may be needed to be included so as to make the database more efficient when managing and querying. The primary keys and foreign keys have deeper background colors to differentiate between other attributes in the tables.

Below is the UML diagram of the Rentals Database


dependents
id int
spouse boolean
children int
infants int
last_updated timestamp
customer_id int
staff
id int
first_name varchar
last_name varchar
gender varchar
position varchar
post_inspection_id int
customer_id int
post_inspection
id int
amenities_after_use_id int
facilities_after_use_id int
entertainment_after_use_id int
staff_id int
customer
id int
first_name varchar
last_name varchar
gender varchar
age int
job_title varchar
job_industry varchar
dependents_id int
address_id int
staff_id int
login_id int
reservation_id int
address
id int
house_no int
street varchar
postcode int
state varchar
last_updated timestamp
city_id int
customer_id int
country
id int
name varchar
continent varchar
last_updated timestamp
login_details
id int
username varchar
password varchar
last_updated datetime
security_questions_id int
login_id int
security_questions
id int
nickname varchar
pet_name varchar
birth_city varchar
login_details_id int
room_address
id int
house_no int
street varchar
postcode int
state varchar
country_id int
room_owner_id int
amenities_id int
room_id int
login
id int
email varchar
phone_number int
create_date datetime
last_login datetime
account_status boolean
account_verified boolean
login_details_id int
customer_id int
room_owner
id int
first_name varchar
last_name varchar
gender varchar
age int
room_address_id int
room_owner_address_id int
reservation
id int
resrvation_type varchar
booking_date datetime
start_date datetime
end_date datetime
last_updated datetime
room_id int
payment_id int
int
room
id int
home_type varchar
total_bedrooms int
total_bathrooms int
published_date datetime
room_address_id int
rules_id int
reservation_id int
facilities
id int
hot_tub boolean
EV_charger boolean
pool boolean
gym boolean
free_parking boolean
entertainment_id int
amenities_id int
amenities
id int
kitchen boolean
air_conditioning boolean
heating boolean
washer boolean
dryer boolean
internet boolean
facilities_id internet
room_address_id internet
rules
id int
pets boolean
smoking boolean
room_id int
payment
id int
mode varchar
price int
discounts int
total_price int
payment_status boolean
reviews_id int
reservation_id int
reviews
id int
stars int
comments varchar
recommend boolean
payment_id int
entertainment_after_use
id int
tv boolean
sound_system boolean
projector boolean
water_front boolean
beach_front boolean
post_inspection_id int
room_owner_address
id int
house_no int
street varchar
state varchar
postcode int
last_updated datetime
country_id int
room_owner_id int
amenities_after_use
id int
kitchen boolean
air_conditioning boolean
heating boolean
washer boolean
dryer boolean
internet boolean
post_inspection_id int
entertainment
id int
tv boolean
sound_system boolean
projector boolean
water_front boolean
beach_front boolean
facilities_id int
facilities_after_use
id int
hot_tub boolean
EV_charger boolean
pool boolean
gym boolean
free_parking boolean
post_inspection_id int
dbdiagram.io