

## **Project Name : Hotel Management And Reservation System**

**Created By: Yangma Lama**

### **Project Description:**

A mini database project built on the idea of a hotel management system using MYSQL and XAMPP. This system will allow users to manage guests, rooms and reservations creating different tables, entities and attributes as per the requirement using different SQL queries.

#### **A. Creating Database and Define tables:**

1. Create a database called "HMS."

```
CREATE DATABASE HMS;
```

2. Create a "Guests" Table.

```
CREATE TABLE Guests (  
    GuestID INT AUTO_INCREMENT PRIMARY KEY,  
    FirstName VARCHAR(50) NOT NULL,  
    LastName VARCHAR(50) NOT NULL,  
    Email VARCHAR(100),  
    PhoneNumber VARCHAR(15)  
);
```

3. Create a "Rooms" Table.

```
CREATE TABLE Rooms (  
    RoomNumber INT AUTO_INCREMENT PRIMARY KEY,  
    RoomType VARCHAR(50) NOT NULL,  
    BedType VARCHAR(50),  
    Rate DECIMAL(10, 2) NOT NULL  
);
```

4. Create a "FoodOrders" Table

```
CREATE TABLE FoodOrders (  
    OrderID INT AUTO_INCREMENT PRIMARY KEY,  
    GuestID INT,  
    OrderDate DATE,  
    TotalAmount DECIMAL(10, 2),  
    FOREIGN KEY (GuestID) REFERENCES Guests(GuestID)  
);
```

5. Create a “Reservations” Table.

```
CREATE TABLE Reservations (  
    ReservationID INT AUTO_INCREMENT PRIMARY KEY,  
    GuestID INT,  
    RoomNumber INT,  
    CheckInDate DATE,  
    CheckOutDate DATE,  
    TotalAmount DECIMAL(10, 2),  
    FOREIGN KEY (GuestID) REFERENCES Guests(GuestID),  
    FOREIGN KEY (RoomNumber) REFERENCES Rooms(RoomNumber)  
);
```

## B. Add data into the database

1. For Guests table

```
-- Insert sample guests  
INSERT INTO Guests (FirstName, LastName, Email, PhoneNumber) VALUES  
( 'John', 'Doe', 'john.doe@gmail.com', '980-456-7890'),  
( 'Will', 'Smith', 'will.smith@yahoo.com', '987-654-3210'),  
( 'Michael', 'Jackson', 'michael.jackson@gmail.com', '986-789-0123'),  
( 'Bob', 'Builder', 'bob.builder@hotmail.com', '984-012-3456'),  
( 'Emily', 'Brown', 'emily.brown@gmail.com', '980-345-6789');
```

GuestID	FirstName	LastName	Email	PhoneNumber
1	John	Doe	john.doe@gmail.com	980-456-7890
2	Will	Smith	will.smith@yahoo.com	987-654-3210
3	Michael	Jackson	michael.jackson@gmail.com	986-789-0123
4	Bob	Builder	bob.builder@gmail.com	984-012-3456
5	Emily	Brown	emily.brown@gmail.com	980-345-6789

2. For Rooms table

```
-- Insert sample rooms  
INSERT INTO Rooms (RoomType, BedType, Rate) VALUES  
( 'Single', 'Queen', 1000.00),  
( 'Double', 'King', 1500.00),  
( 'Suite', 'King', 2500.00),  
( 'Single', 'Twin', 900.00),  
( 'Double', 'Queen', 1600.00);
```

RoomNumber	Room Type	BedType	Rate
1	Single	Queen	1000.00
2	Double	King	1500.00
3	Suite	King	2500.00
4	Single	Twin	900.00
5	Double	Queen	1600.00

3. For Reservation table

```
-- Insert sample reservations
INSERT INTO Reservations (GuestID, RoomNumber, CheckInDate, CheckOutDate, TotalAmount) VALUES
(1, 1, '2024-04-01', '2024-04-05', 4000.00),
(2, 3, '2024-04-03', '2024-04-07', 1000.00),
(3, 2, '2024-04-02', '2024-04-06', 6000.00),
(4, 4, '2024-04-04', '2024-04-08', 4000.00),
(5, 5, '2024-04-05', '2024-04-09', 6400.00);
```

ReservationID	GuestID	RoomNumber	CheckInDate	CheckOutDate	TotalAmount
1	1	1	2024-04-01	2024-04-05	4000.00
2	2	3	2024-04-03	2024-04-07	1000.00
3	3	2	2024-04-02	2024-04-06	6000.00
4	4	4	2024-04-04	2024-04-08	4000.00
5	5	5	2024-04-05	2024-04-09	6400.00

4. For FoodOrders Table

```
-- Insert sample food orders
INSERT INTO FoodOrders (GuestID, OrderDate, TotalAmount) VALUES
(1, '2024-04-01', 800.00),
(2, '2024-04-03', 1200.00),
(3, '2024-04-02', 900.00),
(4, '2024-04-04', 1100.00),
(5, '2024-04-05', 950.00);
```

OrderID	GuestID	OrderDate	TotalAmount
1	1	2024-04-01	800.00
2	2	2024-04-03	1200.00
3	3	2024-04-02	900.00
4	4	2024-04-04	1100.00
5	5	2024-04-05	950.00

## C. SQL Queries:

### 1. Finding Rooms Reserved by a Particular Guest.

```
SELECT RoomNumber, RoomType
FROM Rooms
WHERE RoomNumber IN (
    SELECT RoomNumber
    FROM Reservations
    WHERE GuestID = 1
);
```

RoomNumber	RoomType
1	Single

### 2. Listing All Rooms by Types

```
SELECT RoomType, COUNT(*) AS TotalRooms
FROM Rooms
GROUP BY RoomType;
```

RoomType	TotalRooms
Double	2
Single	2
Suite	1

### 3. Adding “Availability” column to Rooms table

```
ALTER TABLE rooms
ADD Availability bool;
```

RoomNumber	RoomType	BedType	Rate	Availability
1	Single	Queen	1000.00	NULL
2	Double	King	1500.00	NULL
3	Suite	King	2500.00	NULL
4	Single	Twin	900.00	NULL
5	Double	Queen	1600.00	NULL

#### 4. Updating availability of room

```
UPDATE rooms
SET Availability = True
WHERE RoomNumber = 1;

UPDATE rooms
SET Availability = False
WHERE RoomNumber= 2;

UPDATE rooms
SET Availability = True
WHERE RoomNumber = 3;

UPDATE rooms
SET Availability = False
WHERE RoomNumber= 4;
```

RoomNumber	Room Type	BedType	Rate	Availability
1	Single	Queen	1000.00	1
2	Double	King	1500.00	0
3	Suite	King	2500.00	1
4	Single	Twin	900.00	0
5	Double	Queen	1600.00	0

#### 5. Read from the Guest Table

```
-- To read specific columns for a particular guest (e.g., with GuestID = 1)
SELECT FirstName, LastName, Email, PhoneNumber FROM Guests WHERE GuestID = 1;
```

FirstName	LastName	Email	PhoneNumber
John	Doe	john.doe@gmail.com	980-456-7890

```
-- To read specific columns for all guests (e.g., FirstName and LastName)
SELECT FirstName, LastName FROM Guests;
```

FirstName	LastName
John	Doe
Will	Smith
Michael	Jackson
Bob	Builder
Emily	Brown

## 6. Updating information in the Guest Table

```
-- Update a specific guest's email address (e.g., with GuestID = 1)
UPDATE Guests SET Email = 'doe.john@hotmail.com' WHERE GuestID = 1;
```

```
-- Update a specific guest's phone number (e.g., with GuestID = 2)
UPDATE Guests SET PhoneNumber = '981-88-7207' WHERE GuestID = 2;
```

GuestID	FirstName	LastName	Email	PhoneNumber
1	John	Doe	doe.john@hotmail.com	980-456-7890
2	Will	Smith	will.smith@yahoo.com	981-88-7207

## 7. Checking the Availability of a Room using JOIN

```
SELECT r.RoomNumber, r.RoomType
FROM Rooms r
LEFT JOIN Reservations rs ON r.RoomNumber = rs.RoomNumber
WHERE r.Availability IS True;
```

RoomNumber	RoomType
1	Single
3	Suite

## 8. Calculating Number of Days Stayed and Offering 10% Discount for 3 Days (Assuming reservation duration in days):

```
SELECT GuestID,
       DATEDIFF(CheckOutDate, CheckInDate) AS StayDuration,
       CASE
         WHEN DATEDIFF(CheckOutDate, CheckInDate) >= 3 THEN TotalAmount * 0.9
         ELSE TotalAmount
       END AS FinalAmount
FROM Reservations;
```

GuestID	StayDuration	FinalAmount
1	4	3600.000
2	4	900.000
3	1	6000.000
4	4	3600.000
5	3	5760.000

## 9. Calculating 5 % Discount for Total Food amount spent more than 1000.

```
SELECT
    fo.GuestID,
    SUM(fo.TotalAmount) AS TotalFoodAmount,
    CASE
        WHEN SUM(fo.TotalAmount) > 1000 THEN SUM(fo.TotalAmount) * 0.95
        ELSE SUM(fo.TotalAmount)
    END AS FinalFoodAmount
FROM FoodOrders fo
GROUP BY fo.GuestID;
```

GuestID	TotalFoodAmount	FinalFoodAmount
1	800.00	800.0000
2	1200.00	1140.0000
3	900.00	900.0000
4	1100.00	1045.0000
5	950.00	950.0000

## 10. Calculating the Final Amount to be paid by each guest using JOIN.

```
SELECT g.FirstName, g.LastName, SUM(fo.TotalAmount) + SUM(rs.TotalAmount) AS FinalAmount
FROM Guests g
INNER JOIN FoodOrders fo ON g.GuestID = fo.GuestID
INNER JOIN reservations rs ON g.GuestID = rs.GuestID
GROUP BY g.GuestID;
```

FirstName	LastName	FinalAmount
John	Doe	4800.00
Will	Smith	2200.00
Michael	Jackson	6900.00
Bob	Builder	5100.00
Emily	Brown	7350.00