



*Mini project report on*

**Disaster Data Management System**

*Submitted in partial fulfilment of the requirements for the award of degree of*

**Bachelor of Technology**

**in**

**Computer Science & Engineering**

**UE23CS351A – DBMS Project**

*Submitted by:*

**Risu Kumari Nayak**

**PES2UG23CS919**

**Sanjana Saxena**

**PES2UG23CS530**

under the guidance of

**Prof. Nivedita**

Assistant Professor

PES University

**AUG - DEC 2025**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**FACULTY OF ENGINEERING**

**PES UNIVERSITY**

(Established under Karnataka Act No. 16 of 2013)

Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India



## PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)

Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

## CERTIFICATE

*This is to certify that the mini project entitled*

### **HOTEL RESERVATION SYSTEM**

*is a bonafide work carried out by*

**Risu Kumari Nayak**

**PES2UG23CS919**

**Sanjana Saxena**

**PES2UG23CS530**

In partial fulfilment for the completion of fifth semester DBMS Project (UE20CSS301) in the Program of Study -Bachelor of Technology in Computer Science and Engineering under rules and regulations of PES University, Bengaluru during the period AUG. 2022 – DEC. 2022. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The project has been approved as it satisfies the 5<sup>th</sup> semester academic requirements in respect of project work.

Signature

Prof. Nivedita

Assistant Professor

## DECLARATION

We hereby declare that the DBMS Project entitled **HOTEL RESERVATION SYSTEM** has been carried out by us under the guidance of **Prof. Nivedita, Assistant Professor** and submitted in partial fulfilment of the course requirements for the award of degree of **Bachelor of Technology in Computer Science and Engineering** of **PES University, Bengaluru** during the academic semester AUG – DEC 2023.

**Risu Kumari Nayak**

**PES2UG23CS919**     *risu*

**Sanjana Saxena**

**PES2UG23CS530**     *sanjana*

## **ACKNOWLEDGEMENT**

I would like to express my gratitude to Prof.Nivedita, Department of Computer Science and Engineering, PES University, for her continuous guidance, assistance, and encouragement throughout the development of this UE23CS351 - DBMS Project.

I take this opportunity to thank Dr. Sandesh B J, C, Professor,ChairPerson, Department of Computer Science and Engineering, PES University, for all the knowledge and support I have received from the department.

I am deeply grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, Vice-Chancellor, PES University for providing to me various opportunities and enlightenment every step of the way. Finally, this DBMS Project could not have been completed without the continual support and encouragement I have received from my family and friends.

## **ABSTRACT**

The Hotel Reservation System is a database-driven application designed to streamline hotel booking, room allocation, and guest management processes. The system provides a centralized platform that eliminates the inefficiencies of manual reservation handling, reduces booking conflicts, and ensures data accuracy. Developed using MySQL as the database backend and Streamlit for the user interface, the system incorporates essential DBMS concepts including relational schema, ER modeling, triggers, functions, and stored procedures. It supports multiple hotels, room types, and guests, and enables administrators to manage reservations in real time. Features such as automated room status updates, secure admin login, and payment logging improve operational efficiency and maintain data integrity. Overall, this project demonstrates a comprehensive and scalable approach to hotel management automation, highlighting effective utilization of database technologies and modern UI integration.

# **TABLE OF CONTENTS**

<b>Chapter No.</b>	<b>Title</b>	<b>Page No.</b>
1.	INTRODUCTION	7
2.	PROBLEM DEFINITION	8
3.	ER MODEL	9
4.	ER TO RELATIONAL MAPPING	10
5.	DDL STATEMENTS	12
6.	DML STATEMENTS	17
7.	QUERIES (SIMPLE QUERY AND UPDATE AND DELETE OPERATION, CORRELATED QUERY AND NESTED QUERY)	26
8.	STORED PROCEDURE, FUNCTIONS AND TRIGGERS	31
9.	FRONT END DEVELOPMENT	34
	REFERENCES/BIBLIOGRAPHY	38
	APPENDIX A DEFINITIONS, ACRONYMS AND ABBREVIATIONS	39
	GITHUB REPO LINK	40

# 1. INTRODUCTION

This project presents a **Hotel Reservation System** designed using **MySQL** as the backend database and **Python Flask along with HTML** as the user interface.

The system automates:

- Room booking
- Guest management
- Reservation tracking
- Payment logging
- Room status updates

This project demonstrates DBMS concepts such as:

- ER Diagrams
- Relational schema
- Triggers
- Stored procedures
- Functions
- Transactions
- Front-end integration

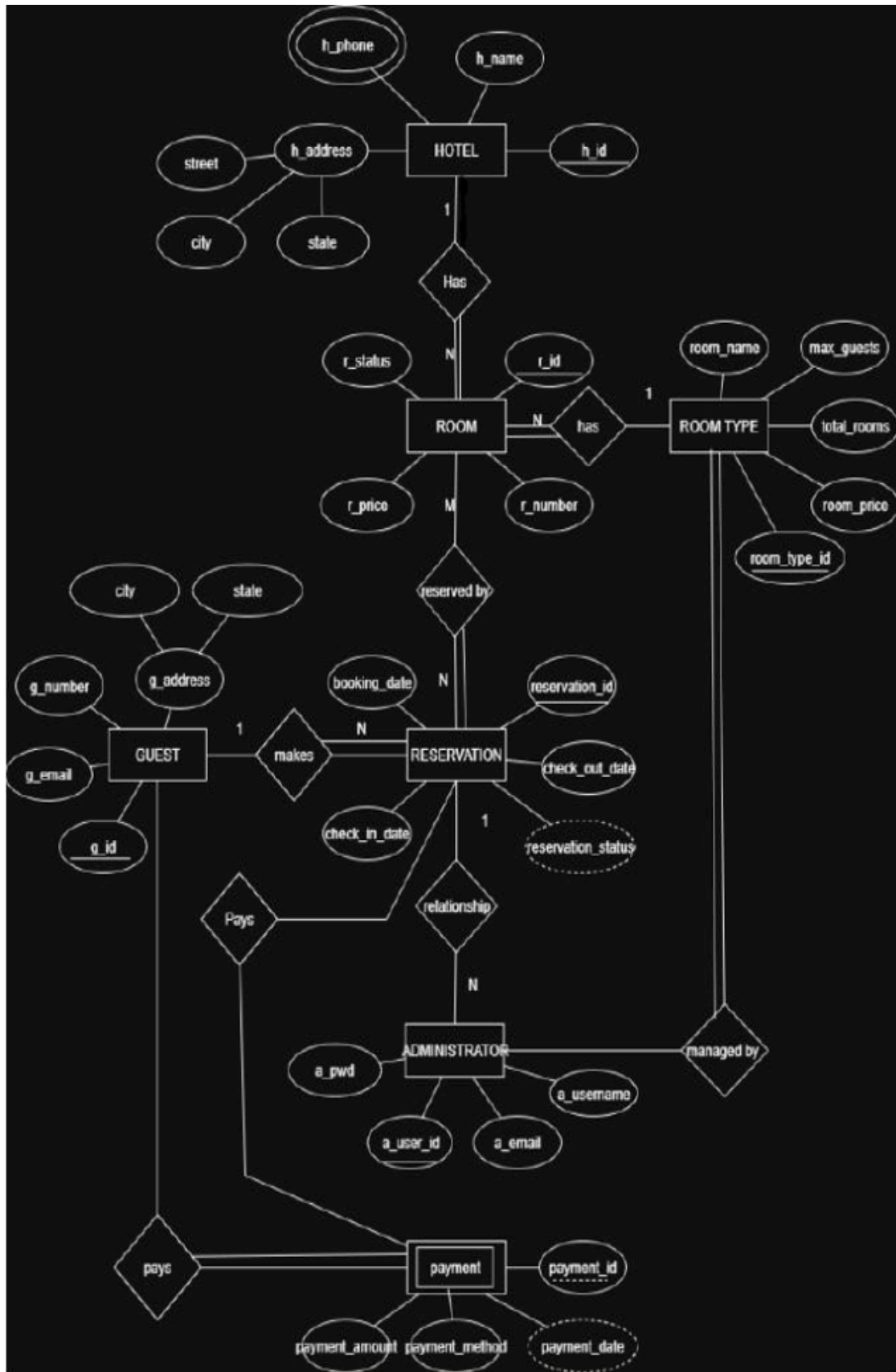
## **2. PROBLEM DEFINITION**

Manual hotel reservation systems often suffer from:

- Double booking
- Data duplication
- Difficulty in tracking room availability
- No centralized control
- Human errors

The goal of this project is to build a centralized and automated reservation system that provides accurate, real-time updates and supports hotel administrators in managing bookings efficiently.

### 3. ER MODEL

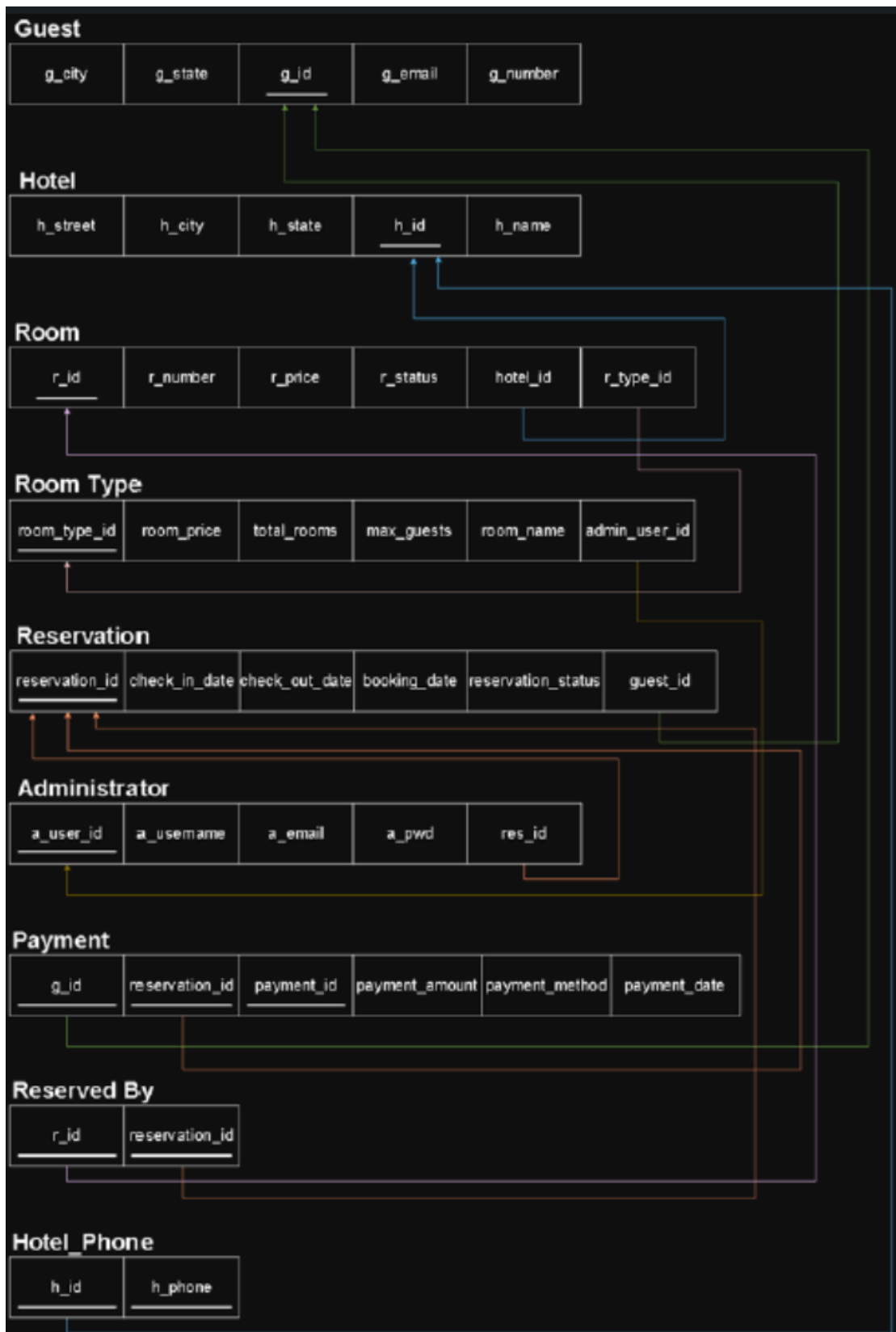


## **4. ER TO RELATIONAL MAPPING**

### **4.1 STEPS OF ALGORITHM FOR CHOSEN PROBLEM**

1. Mapping Strong Entities
2. Mapping Weak Entities
3. Mapping 1:N Relationships
4. Mapping M:N Relationships
5. Mapping Multivalued Attributes
6. Mapping Composite Attributes
7. Mapping Derived Attributes

## 4.2 COMPLETE DIAGRAM OF RELATIONAL MAPPING



## 5. DDL STATEMENTS

```
mysql> CREATE DATABASE Hotel_Management_System;
Query OK, 1 row affected (0.03 sec)
```

```
mysql> USE Hotel_Management_System;
Database changed
```

```
mysql> CREATE TABLE Guest (
  ->   g_id INT PRIMARY KEY AUTO_INCREMENT,
  ->   g_name VARCHAR(50),
  ->   g_email VARCHAR(100) UNIQUE,
  ->   g_number VARCHAR(15),
  ->   g_city VARCHAR(50),
  ->   g_state VARCHAR(50)
  -> );
```

```
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> DESCRIBE Guest;
```

Field	Type	Null	Key	Default	Extra
g_id	int	NO	PRI	NULL	auto_increment
g_name	varchar(50)	YES		NULL	
g_email	varchar(100)	YES	UNI	NULL	
g_number	varchar(15)	YES		NULL	
g_city	varchar(50)	YES		NULL	
g_state	varchar(50)	YES		NULL	

```
6 rows in set (0.06 sec)
```

```
mysql>
mysql> CREATE TABLE Hotel (
  ->   h_id INT PRIMARY KEY AUTO_INCREMENT,
  ->   h_name VARCHAR(100),
  ->   h_street VARCHAR(100),
  ->   h_city VARCHAR(50),
  ->   h_state VARCHAR(50)
  -> );
```

```
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> DESCRIBE Hotel;
```

Field	Type	Null	Key	Default	Extra
h_id	int	NO	PRI	NULL	auto_increment
h_name	varchar(100)	YES		NULL	
h_street	varchar(100)	YES		NULL	
h_city	varchar(50)	YES		NULL	
h_state	varchar(50)	YES		NULL	

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

```
mysql>
mysql> CREATE TABLE Room_Type (
  ->   room_type_id INT PRIMARY KEY AUTO_INCREMENT,
  ->   room_name VARCHAR(50),
  ->   room_price DECIMAL(10,2),
  ->   total_rooms INT,
  ->   max_guests INT,
  ->   admin_user_id INT
```

```

    ->     admin_user_id INT
    -> );
Query OK, 0 rows affected (0.02 sec)

```

```
mysql> DESCRIBE Room_Type;
```

Field	Type	Null	Key	Default	Extra
room_type_id	int	NO	PRI	NULL	auto_increment
room_name	varchar(50)	YES		NULL	
room_price	decimal(10,2)	YES		NULL	
total_rooms	int	YES		NULL	
max_guests	int	YES		NULL	
admin_user_id	int	YES		NULL	

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

```

mysql>
mysql> CREATE TABLE Room (
    ->     r_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     r_number VARCHAR(10),
    ->     r_price DECIMAL(10,2),
    ->     r_status VARCHAR(20),
    ->     hotel_id INT,
    ->     r_type_id INT,
    ->     FOREIGN KEY (hotel_id) REFERENCES Hotel(h_id),
    ->     FOREIGN KEY (r_type_id) REFERENCES Room_Type(room_type_id)
    -> );

```

```
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> DESCRIBE Room;
```

Field	Type	Null	Key	Default	Extra
r_id	int	NO	PRI	NULL	auto_increment
r_number	varchar(10)	YES		NULL	
r_price	decimal(10,2)	YES		NULL	
r_status	varchar(20)	YES		NULL	
hotel_id	int	YES	MUL	NULL	
r_type_id	int	YES	MUL	NULL	

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

```

mysql>
mysql> CREATE TABLE Reservation (
    ->     reservation_id INT PRIMARY KEY AUTO_INCREMENT,
    ->     guest_id INT,
    ->     check_in_date DATE,
    ->     check_out_date DATE,
    ->     booking_date DATE,
    ->     reservation_status VARCHAR(20),
    ->     FOREIGN KEY (guest_id) REFERENCES Guest(g_id)
    -> );

```

```

-> FOREIGN KEY (guest_id) REFERENCES Guest(g_id)
-> );
Query OK, 0 rows affected (0.04 sec)

```

```
mysql> DESCRIBE Reservation;
```

Field	Type	Null	Key	Default	Extra
reservation_id	int	NO	PRI	NULL	auto_increment
guest_id	int	YES	MUL	NULL	
check_in_date	date	YES		NULL	
check_out_date	date	YES		NULL	
booking_date	date	YES		NULL	
reservation_status	varchar(20)	YES		NULL	

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

```
mysql>
```

```
mysql> CREATE TABLE Administrator (
-> a_user_id INT PRIMARY KEY AUTO_INCREMENT,
-> a_username VARCHAR(50) UNIQUE,
-> a_email VARCHAR(100),
-> a_pwd VARCHAR(50),
-> res_id INT,
-> FOREIGN KEY (res_id) REFERENCES Reservation(reservation_id)
-> );
Query OK, 0 rows affected (0.03 sec)

```

```
mysql> DESCRIBE Administrator;
```

Field	Type	Null	Key	Default	Extra
a_user_id	int	NO	PRI	NULL	auto_increment
a_username	varchar(50)	YES	UNI	NULL	
a_email	varchar(100)	YES		NULL	
a_pwd	varchar(50)	YES		NULL	
res_id	int	YES	MUL	NULL	

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

```
mysql>
```

```
mysql> CREATE TABLE Payment (
-> payment_id INT PRIMARY KEY AUTO_INCREMENT,
-> g_id INT,
-> reservation_id INT,
-> payment_amount DECIMAL(10,2),
-> payment_method VARCHAR(30),
-> payment_date DATE,
-> FOREIGN KEY (g_id) REFERENCES Guest(g_id),
-> FOREIGN KEY (reservation_id) REFERENCES Reservation(reservation_id)
-> );
Query OK, 0 rows affected (0.06 sec)

```

```

    -> FOREIGN KEY (reservation_id) REFERENCES Reservation(reservation_id)
    -> );
Query OK, 0 rows affected (0.06 sec)

```

```
mysql> DESCRIBE Payment;
```

Field	Type	Null	Key	Default	Extra
payment_id	int	NO	PRI	NULL	auto_increment
g_id	int	YES	MUL	NULL	
reservation_id	int	YES	MUL	NULL	
payment_amount	decimal(10,2)	YES		NULL	
payment_method	varchar(30)	YES		NULL	
payment_date	date	YES		NULL	

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

```
mysql>
```

```
mysql> CREATE TABLE Reserved_By (
```

```

    -> r_id INT,
    -> reservation_id INT,
    -> PRIMARY KEY (r_id, reservation_id),
    -> FOREIGN KEY (r_id) REFERENCES Room(r_id),
    -> FOREIGN KEY (reservation_id) REFERENCES Reservation(reservation_id)
    -> );

```

```
Query OK, 0 rows affected (0.05 sec)
```

```
mysql> DESCRIBE Reserved_By;
```

Field	Type	Null	Key	Default	Extra
r_id	int	NO	PRI	NULL	
reservation_id	int	NO	PRI	NULL	

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

```
mysql>
```

```
mysql> CREATE TABLE Hotel_Phone (
```

```

    -> h_id INT,
    -> h_phone VARCHAR(15),
    -> PRIMARY KEY (h_id, h_phone),
    -> FOREIGN KEY (h_id) REFERENCES Hotel(h_id)
    -> );

```

```
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> DESCRIBE Hotel_Phone;
```

Field	Type	Null	Key	Default	Extra
h_id	int	NO	PRI	NULL	
h_phone	varchar(15)	NO	PRI	NULL	

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

```
mysql> CREATE TABLE Payment_Log (
->     log_id INT AUTO_INCREMENT PRIMARY KEY,
->     payment_id INT,
->     log_message VARCHAR(255),
->     log_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
-> );
```

Query OK, 0 rows affected (0.03 sec)

```
mysql> DESCRIBE Payment_Log;
```

Field	Type	Null	Key	Default	Extra
log_id	int	NO	PRI	NULL	auto_increment
payment_id	int	YES		NULL	
log_message	varchar(255)	YES		NULL	
log_time	timestamp	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED

4 rows in set (0.00 sec)

## 6. DML STATEMENTS

```
mysql> INSERT INTO Hotel (h_name, h_street, h_city, h_state) VALUES
-> ('Taj Palace', 'MG Road', 'Delhi', 'Delhi'),
-> ('Oberoi Grand', 'Park Street', 'Kolkata', 'West Bengal'),
-> ('The Leela Palace', 'Bhikaji Cama Place', 'New Delhi', 'Delhi'),
-> ('ITC Maurya', 'Sardar Patel Marg', 'New Delhi', 'Delhi'),
-> ('Radisson Blu', 'Mahipalpur', 'New Delhi', 'Delhi'),
-> ('Hilton Mumbai', 'Nariman Point', 'Mumbai', 'Maharashtra'),
-> ('Marriott Bangalore', 'Whitefield', 'Bangalore', 'Karnataka'),
-> ('Hyatt Bangalore', 'Koramangala', 'Bangalore', 'Karnataka');
```

```
Query OK, 8 rows affected (0.00 sec)
Records: 8 Duplicates: 0 Warnings: 0
```

```
mysql>
mysql> SELECT * FROM Hotel;
```

h_id	h_name	h_street	h_city	h_state
1	Taj Palace	MG Road	Delhi	Delhi
2	Oberoi Grand	Park Street	Kolkata	West Bengal
3	The Leela Palace	Bhikaji Cama Place	New Delhi	Delhi
4	ITC Maurya	Sardar Patel Marg	New Delhi	Delhi
5	Radisson Blu	Mahipalpur	New Delhi	Delhi
6	Hilton Mumbai	Nariman Point	Mumbai	Maharashtra
7	Marriott Bangalore	Whitefield	Bangalore	Karnataka
8	Hyatt Bangalore	Koramangala	Bangalore	Karnataka

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

```
mysql>
mysql> -- =====
mysql> -- INSERT ROOM TYPES
mysql> -- =====
mysql>
```

```
mysql> INSERT INTO Room_Type (room_name, room_price, total_rooms, max_guests, admin_user_id) VALUES
-> ('Single Room', 2500, 30, 1, NULL),
-> ('Deluxe', 4000, 40, 2, NULL),
-> ('Double Bed', 4500, 35, 2, NULL),
-> ('Suite', 7000, 20, 4, NULL),
-> ('Presidential Suite', 15000, 5, 6, NULL),
-> ('Twin Bed', 3500, 25, 2, NULL),
-> ('Family Room', 6000, 15, 4, NULL);
```

```
Query OK, 7 rows affected (0.00 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

```
mysql>
mysql> SELECT * FROM Room_Type;
```

room_type_id	room_name	room_price	total_rooms	max_guests	admin_user_id
1	Single Room	2500.00	30	1	NULL
2	Deluxe	4000.00	40	2	NULL
3	Double Bed	4500.00	35	2	NULL
4	Suite	7000.00	20	4	NULL

3	Double Bed	4500.00	35	2	NULL
4	Suite	7000.00	20	4	NULL
5	Presidential Suite	15000.00	5	6	NULL
6	Twin Bed	3500.00	25	2	NULL
7	Family Room	6000.00	15	4	NULL

7 rows in set (0.00 sec)

mysql>

mysql> -- =====

mysql> -- INSERT ROOMS FOR HOTELS

mysql> -- =====

mysql>

mysql> INSERT INTO Room (r\_number, r\_price, r\_status, hotel\_id, r\_type\_id) VALUES

```

-> ('S101', 2500, 'Available', 1, 1),
-> ('S102', 2500, 'Available', 1, 1),
-> ('S103', 2500, 'Available', 1, 1),
-> ('S104', 2500, 'Available', 1, 1),
-> ('S105', 2500, 'Available', 1, 1),
-> ('D201', 4000, 'Available', 1, 2),
-> ('D202', 4000, 'Available', 1, 2),
-> ('D203', 4000, 'Available', 1, 2),
-> ('D204', 4000, 'Available', 1, 2),
-> ('D205', 4000, 'Available', 1, 2),
-> ('DB301', 4500, 'Available', 1, 3),
-> ('DB302', 4500, 'Available', 1, 3),
-> ('DB303', 4500, 'Available', 1, 3),
-> ('SU401', 7000, 'Available', 1, 4),
-> ('SU402', 7000, 'Available', 1, 4),
-> ('PS501', 15000, 'Available', 1, 5);

```

Query OK, 16 rows affected (0.01 sec)

Records: 16 Duplicates: 0 Warnings: 0

mysql>

mysql> SELECT \* FROM Room WHERE hotel\_id = 1;

r_id	r_number	r_price	r_status	hotel_id	r_type_id
1	S101	2500.00	Available	1	1
2	S102	2500.00	Available	1	1
3	S103	2500.00	Available	1	1
4	S104	2500.00	Available	1	1
5	S105	2500.00	Available	1	1
6	D201	4000.00	Available	1	2
7	D202	4000.00	Available	1	2
8	D203	4000.00	Available	1	2
9	D204	4000.00	Available	1	2
10	D205	4000.00	Available	1	2
11	DB301	4500.00	Available	1	3
12	DB302	4500.00	Available	1	3
13	DB303	4500.00	Available	1	3
14	SU401	7000.00	Available	1	4
15	SU402	7000.00	Available	1	4

13	DB303	4500.00	Available	1	3
14	SU401	7000.00	Available	1	4
15	SU402	7000.00	Available	1	4
16	PS501	15000.00	Available	1	5

16 rows in set (0.00 sec)

mysql>

mysql> INSERT INTO Room (r\_number, r\_price, r\_status, hotel\_id, r\_type\_id) VALUES

```

-> ('S101', 2800, 'Available', 2, 1),
-> ('S102', 2800, 'Available', 2, 1),
-> ('S103', 2800, 'Available', 2, 1),
-> ('S104', 2800, 'Available', 2, 1),
-> ('D201', 4500, 'Available', 2, 2),
-> ('D202', 4500, 'Available', 2, 2),
-> ('D203', 4500, 'Available', 2, 2),
-> ('D204', 4500, 'Available', 2, 2),
-> ('T301', 3800, 'Available', 2, 6),
-> ('T302', 3800, 'Available', 2, 6),
-> ('F401', 6500, 'Available', 2, 7),
-> ('F402', 6500, 'Available', 2, 7);

```

Query OK, 12 rows affected (0.00 sec)

Records: 12 Duplicates: 0 Warnings: 0

mysql>

mysql> SELECT \* FROM Room WHERE hotel\_id = 2;

r_id	r_number	r_price	r_status	hotel_id	r_type_id
17	S101	2800.00	Available	2	1
18	S102	2800.00	Available	2	1
19	S103	2800.00	Available	2	1
20	S104	2800.00	Available	2	1
21	D201	4500.00	Available	2	2
22	D202	4500.00	Available	2	2
23	D203	4500.00	Available	2	2
24	D204	4500.00	Available	2	2
25	T301	3800.00	Available	2	6
26	T302	3800.00	Available	2	6
27	F401	6500.00	Available	2	7
28	F402	6500.00	Available	2	7

12 rows in set (0.00 sec)

mysql>

mysql> INSERT INTO Room (r\_number, r\_price, r\_status, hotel\_id, r\_type\_id) VALUES

```

-> ('S101', 3200, 'Available', 3, 1),
-> ('S102', 3200, 'Available', 3, 1),
-> ('S103', 3200, 'Available', 3, 1),
-> ('D201', 5500, 'Available', 3, 2),
-> ('D202', 5500, 'Available', 3, 2),
-> ('D203', 5500, 'Available', 3, 2),
-> ('D204', 5500, 'Available', 3, 2),

```

```

-> ('D203', 5500, 'Available', 3, 2),
-> ('D204', 5500, 'Available', 3, 2),
-> ('D205', 5500, 'Available', 3, 2),
-> ('DB301', 5200, 'Available', 3, 3),
-> ('DB302', 5200, 'Available', 3, 3),
-> ('SU401', 8500, 'Available', 3, 4),
-> ('SU402', 8500, 'Available', 3, 4),
-> ('SU403', 8500, 'Available', 3, 4);

```

Query OK, 13 rows affected (0.00 sec)

Records: 13 Duplicates: 0 Warnings: 0

mysql>

mysql> SELECT \* FROM Room WHERE hotel\_id = 3;

r_id	r_number	r_price	r_status	hotel_id	r_type_id
29	S101	3200.00	Available	3	1
30	S102	3200.00	Available	3	1
31	S103	3200.00	Available	3	1
32	D201	5500.00	Available	3	2
33	D202	5500.00	Available	3	2
34	D203	5500.00	Available	3	2
35	D204	5500.00	Available	3	2
36	D205	5500.00	Available	3	2
37	DB301	5200.00	Available	3	3
38	DB302	5200.00	Available	3	3
39	SU401	8500.00	Available	3	4
40	SU402	8500.00	Available	3	4
41	SU403	8500.00	Available	3	4

13 rows in set (0.00 sec)

mysql>

mysql> INSERT INTO Room (r\_number, r\_price, r\_status, hotel\_id, r\_type\_id) VALUES

```

-> ('S101', 3000, 'Available', 4, 1),
-> ('S102', 3000, 'Available', 4, 1),
-> ('S103', 3000, 'Available', 4, 1),
-> ('S104', 3000, 'Available', 4, 1),
-> ('D201', 5000, 'Available', 4, 2),
-> ('D202', 5000, 'Available', 4, 2),
-> ('D203', 5000, 'Available', 4, 2),
-> ('D204', 5000, 'Available', 4, 2),
-> ('DB301', 4800, 'Available', 4, 3),
-> ('DB302', 4800, 'Available', 4, 3),
-> ('DB303', 4800, 'Available', 4, 3),
-> ('F401', 7200, 'Available', 4, 7),
-> ('F402', 7200, 'Available', 4, 7);

```

Query OK, 13 rows affected (0.00 sec)

Records: 13 Duplicates: 0 Warnings: 0

mysql>

mysql> SELECT \* FROM Room WHERE hotel\_id = 4;

r_id	r_number	r_price	r_status	hotel_id	r_type_id
------	----------	---------	----------	----------	-----------

```
mysql> SELECT * FROM Room WHERE hotel_id = 4;
```

r_id	r_number	r_price	r_status	hotel_id	r_type_id
42	S101	3000.00	Available	4	1
43	S102	3000.00	Available	4	1
44	S103	3000.00	Available	4	1
45	S104	3000.00	Available	4	1
46	D201	5000.00	Available	4	2
47	D202	5000.00	Available	4	2
48	D203	5000.00	Available	4	2
49	D204	5000.00	Available	4	2
50	DB301	4800.00	Available	4	3
51	DB302	4800.00	Available	4	3
52	DB303	4800.00	Available	4	3
53	F401	7200.00	Available	4	7
54	F402	7200.00	Available	4	7

13 rows in set (0.00 sec)

```
mysql>
```

```
mysql> INSERT INTO Room (r_number, r_price, r_status, hotel_id, r_type_id) VALUES
-> ('S101', 2200, 'Available', 5, 1),
-> ('S102', 2200, 'Available', 5, 1),
-> ('S103', 2200, 'Available', 5, 1),
-> ('S104', 2200, 'Available', 5, 1),
-> ('S105', 2200, 'Available', 5, 1),
-> ('D201', 3500, 'Available', 5, 2),
-> ('D202', 3500, 'Available', 5, 2),
-> ('D203', 3500, 'Available', 5, 2),
-> ('D204', 3500, 'Available', 5, 2),
-> ('T301', 3200, 'Available', 5, 6),
-> ('T302', 3200, 'Available', 5, 6),
-> ('T303', 3200, 'Available', 5, 6);
```

Query OK, 12 rows affected (0.00 sec)

Records: 12 Duplicates: 0 Warnings: 0

```
mysql>
```

```
mysql> SELECT * FROM Room WHERE hotel_id = 5;
```

r_id	r_number	r_price	r_status	hotel_id	r_type_id
55	S101	2200.00	Available	5	1
56	S102	2200.00	Available	5	1
57	S103	2200.00	Available	5	1
58	S104	2200.00	Available	5	1
59	S105	2200.00	Available	5	1
60	D201	3500.00	Available	5	2
61	D202	3500.00	Available	5	2
62	D203	3500.00	Available	5	2
63	D204	3500.00	Available	5	2
64	T301	3200.00	Available	5	6

63	D204	3500.00	Available	5	2
64	T301	3200.00	Available	5	6
65	T302	3200.00	Available	5	6
66	T303	3200.00	Available	5	6

12 rows in set (0.00 sec)

mysql>

```
mysql> INSERT INTO Room (r_number, r_price, r_status, hotel_id, r_type_id) VALUES
-> ('S101', 3500, 'Available', 6, 1),
-> ('S102', 3500, 'Available', 6, 1),
-> ('S103', 3500, 'Available', 6, 1),
-> ('D201', 5500, 'Available', 6, 2),
-> ('D202', 5500, 'Available', 6, 2),
-> ('D203', 5500, 'Available', 6, 2),
-> ('D204', 5500, 'Available', 6, 2),
-> ('DB301', 5200, 'Available', 6, 3),
-> ('DB302', 5200, 'Available', 6, 3),
-> ('SU401', 9000, 'Available', 6, 4),
-> ('SU402', 9000, 'Available', 6, 4);
```

Query OK, 11 rows affected (0.00 sec)

Records: 11 Duplicates: 0 Warnings: 0

mysql>

```
mysql> SELECT * FROM Room WHERE hotel_id = 6;
```

r_id	r_number	r_price	r_status	hotel_id	r_type_id
67	S101	3500.00	Available	6	1
68	S102	3500.00	Available	6	1
69	S103	3500.00	Available	6	1
70	D201	5500.00	Available	6	2
71	D202	5500.00	Available	6	2
72	D203	5500.00	Available	6	2
73	D204	5500.00	Available	6	2
74	DB301	5200.00	Available	6	3
75	DB302	5200.00	Available	6	3
76	SU401	9000.00	Available	6	4
77	SU402	9000.00	Available	6	4

11 rows in set (0.00 sec)

mysql>

```
mysql> INSERT INTO Room (r_number, r_price, r_status, hotel_id, r_type_id) VALUES
-> ('S101', 2800, 'Available', 7, 1),
-> ('S102', 2800, 'Available', 7, 1),
-> ('S103', 2800, 'Available', 7, 1),
-> ('D201', 4200, 'Available', 7, 2),
-> ('D202', 4200, 'Available', 7, 2),
-> ('D203', 4200, 'Available', 7, 2),
-> ('D204', 4200, 'Available', 7, 2),
-> ('D205', 4200, 'Available', 7, 2),
-> ('T301', 3800, 'Available', 7, 6),
```

```

-> ('T301', 3800, 'Available', 7, 6),
-> ('T302', 3800, 'Available', 7, 6),
-> ('F401', 6200, 'Available', 7, 7);

```

Query OK, 11 rows affected (0.00 sec)  
Records: 11 Duplicates: 0 Warnings: 0

```

mysql>
mysql> SELECT * FROM Room WHERE hotel_id = 7;

```

r_id	r_number	r_price	r_status	hotel_id	r_type_id
78	S101	2800.00	Available	7	1
79	S102	2800.00	Available	7	1
80	S103	2800.00	Available	7	1
81	D201	4200.00	Available	7	2
82	D202	4200.00	Available	7	2
83	D203	4200.00	Available	7	2
84	D204	4200.00	Available	7	2
85	D205	4200.00	Available	7	2
86	T301	3800.00	Available	7	6
87	T302	3800.00	Available	7	6
88	F401	6200.00	Available	7	7

11 rows in set (0.00 sec)

```

mysql>
mysql> INSERT INTO Room (r_number, r_price, r_status, hotel_id, r_type_id) VALUES

```

```

-> ('S101', 3000, 'Available', 8, 1),
-> ('S102', 3000, 'Available', 8, 1),
-> ('D201', 4800, 'Available', 8, 2),
-> ('D202', 4800, 'Available', 8, 2),
-> ('D203', 4800, 'Available', 8, 2),
-> ('D204', 4800, 'Available', 8, 2),
-> ('DB301', 5000, 'Available', 8, 3),
-> ('DB302', 5000, 'Available', 8, 3),
-> ('SU401', 8200, 'Available', 8, 4),
-> ('SU402', 8200, 'Available', 8, 4),
-> ('F401', 6800, 'Available', 8, 7);

```

Query OK, 11 rows affected (0.00 sec)  
Records: 11 Duplicates: 0 Warnings: 0

```

mysql>
mysql> SELECT * FROM Room WHERE hotel_id = 8;

```

r_id	r_number	r_price	r_status	hotel_id	r_type_id
89	S101	3000.00	Available	8	1
90	S102	3000.00	Available	8	1
91	D201	4800.00	Available	8	2
92	D202	4800.00	Available	8	2
93	D203	4800.00	Available	8	2
94	D204	4800.00	Available	8	2
95	DB301	5000.00	Available	8	3

95	DB301	5000.00	Available	8	3
96	DB302	5000.00	Available	8	3
97	SU401	8200.00	Available	8	4
98	SU402	8200.00	Available	8	4
99	F401	6800.00	Available	8	7

11 rows in set (0.00 sec)

mysql>

mysql> -- =====

mysql> -- INSERT HOTEL PHONE NUMBERS

mysql> -- =====

mysql>

mysql> INSERT INTO Hotel\_Phone (h\_id, h\_phone) VALUES

-> (1, '011-23456789'),

-> (1, '011-23456790'),

-> (2, '033-22876543'),

-> (2, '033-22876544'),

-> (3, '011-41234567'),

-> (3, '011-41234568'),

-> (4, '011-26117777'),

-> (4, '011-26117778'),

-> (5, '011-41657777'),

-> (5, '011-41657778'),

-> (6, '022-61322323'),

-> (6, '022-61322324'),

-> (7, '080-41234567'),

-> (7, '080-41234568'),

-> (8, '080-41515151'),

-> (8, '080-41515152');

Query OK, 16 rows affected (0.00 sec)

Records: 16 Duplicates: 0 Warnings: 0

mysql>

mysql> SELECT \* FROM Hotel\_Phone;

h_id	h_phone
1	011-23456789
1	011-23456790
2	033-22876543
2	033-22876544
3	011-41234567
3	011-41234568
4	011-26117777
4	011-26117778
5	011-41657777
5	011-41657778
6	022-61322323
6	022-61322324
7	080-41234567
7	080-41234568
8	080-41515151

7	080-41234568
8	080-41515151
8	080-41515152

16 rows in set (0.00 sec)

mysql>

mysql> -- =====

mysql> -- INSERT ADMIN USERS

mysql> -- =====

mysql>

mysql> INSERT INTO Administrator (a\_username, a\_email, a\_pwd, res\_id) VALUES

-> ('admin\_taj', 'admin@tajpalace.com', 'admin123', NULL),

-> ('admin\_oberoi', 'admin@oberoi.com', 'admin123', NULL),

-> ('admin\_leela', 'admin@leela.com', 'admin123', NULL),

-> ('admin\_itc', 'admin@itc.com', 'admin123', NULL),

-> ('admin\_radisson', 'admin@radisson.com', 'admin123', NULL),

-> ('admin\_hilton', 'admin@hilton.com', 'admin123', NULL),

-> ('admin\_marriott', 'admin@marriott.com', 'admin123', NULL),

-> ('admin\_hyatt', 'admin@hyatt.com', 'admin123', NULL);

Query OK, 8 rows affected (0.00 sec)

Records: 8 Duplicates: 0 Warnings: 0

mysql>

mysql> SELECT \* FROM Administrator;

	a_user_id	a_username	a_email	a_pwd	res_id
	1	admin_taj	admin@tajpalace.com	admin123	NULL
	2	admin_oberoi	admin@oberoi.com	admin123	NULL
	3	admin_leela	admin@leela.com	admin123	NULL
	4	admin_itc	admin@itc.com	admin123	NULL
	5	admin_radisson	admin@radisson.com	admin123	NULL
	6	admin_hilton	admin@hilton.com	admin123	NULL
	7	admin_marriott	admin@marriott.com	admin123	NULL
	8	admin_hyatt	admin@hyatt.com	admin123	NULL

8 rows in set (0.00 sec)

## 7. QUERIES

### 7.1 SIMPLE QUERY WITH GROUP BY, AGGREGATE

```
mysql> SELECT h.h_name, COUNT(r.r_id) AS total_rooms  
-> FROM Hotel h  
-> LEFT JOIN Room r ON h.h_id = r.hotel_id  
-> GROUP BY h.h_id, h.h_name;
```

h_name	total_rooms
Taj Palace	16
Oberoi Grand	12
The Leela Palace	13
ITC Maurya	13
Radisson Blu	12
Hilton Mumbai	11
Marriott Bangalore	11
Hyatt Bangalore	11

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

## 7.2 UPDATE OPERATION

```
mysql> UPDATE Room
      -> SET r_status = 'Booked'
      -> WHERE r_id = 1;
Query OK, 1 row affected (0.13 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql>
mysql> SELECT * FROM Room WHERE r_id = 1;
+-----+-----+-----+-----+-----+-----+
| r_id | r_number | r_price | r_status | hotel_id | r_type_id |
+-----+-----+-----+-----+-----+-----+
|    1 | S101    | 2500.00 | Booked   |         1 |          1 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

## 7.3 DELETE OPERATION

```
mysql> DELETE FROM Hotel_Phone
      -> WHERE h_phone = '011-23456790';
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> SELECT * FROM Hotel_Phone WHERE h_id = 1;
+-----+-----+
| h_id | h_phone |
+-----+-----+
|    1 | 011-23456789 |
+-----+-----+
1 row in set (0.00 sec)
```

## 7.4 CORRELATED QUERY

```
mysql> INSERT INTO Guest (g_name, g_email, g_number, g_city, g_state)
-> VALUES
-> ('Riya Sharma', 'riya@gmail.com', '9876543210', 'Delhi', 'Delhi'),
-> ('Aditi Mehta', 'aditi@gmail.com', '9988776655', 'Mumbai', 'Maharashtra');
Query OK, 2 rows affected (0.01 sec)
Records: 2 Duplicates: 0 Warnings: 0

mysql>
mysql> SELECT * FROM Guest;
+-----+-----+-----+-----+-----+-----+
| g_id | g_name   | g_email      | g_number    | g_city  | g_state  |
+-----+-----+-----+-----+-----+-----+
| 1    | Riya Sharma | riya@gmail.com | 9876543210 | Delhi   | Delhi    |
| 2    | Aditi Mehta | aditi@gmail.com | 9988776655 | Mumbai  | Maharashtra |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
mysql> -- Insert sample reservations
mysql> INSERT INTO Reservation (guest_id, check_in_date, check_out_date, booking_date, reservation_status)
-> VALUES
-> (1, '2025-11-10', '2025-11-12', CURDATE(), 'Booked'),
-> (2, '2025-12-01', '2025-12-03', CURDATE(), 'Booked');
Query OK, 2 rows affected (0.00 sec)
Records: 2 Duplicates: 0 Warnings: 0

mysql>
mysql> SELECT * FROM Reservation;
+-----+-----+-----+-----+-----+-----+
| reservation_id | guest_id | check_in_date | check_out_date | booking_date | reservation_status |
+-----+-----+-----+-----+-----+-----+
| 1              | 1        | 2025-11-10    | 2025-11-12    | 2025-11-10   | Booked             |
| 2              | 2        | 2025-12-01    | 2025-12-03    | 2025-11-10   | Booked             |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> SELECT g.g_name
-> FROM Guest g
-> WHERE EXISTS (
->     SELECT 1
->     FROM Reservation r
->     WHERE r.guest_id = g.g_id
-> );
+-----+
| g_name |
+-----+
| Riya Sharma |
| Aditi Mehta |
+-----+
2 rows in set (0.01 sec)
```

## 7.5 NESTED QUERY

```
mysql> SELECT h_name  
-> FROM Hotel  
-> WHERE h_id IN (  
->     SELECT hotel_id  
->     FROM Room  
->     WHERE r_price > (SELECT AVG(r_price) FROM Room)  
-> );
```

h_name
Taj Palace
Oberoi Grand
The Leela Palace
ITC Maurya
Hilton Mumbai
Marriott Bangalore
Hyatt Bangalore

7 rows in set (0.02 sec)

## 8. STORED PROCEDURES, FUNCTIONS AND TRIGGERS

### 8.1 STORED PROCEDURES OR FUNCTIONS

```
mysql> DELIMITER $$
mysql> CREATE PROCEDURE sp_BookRoom(
->     IN p_guest_id INT,
->     IN p_room_id INT,
->     IN p_checkin DATE,
->     IN p_checkout DATE
-> )
-> BEGIN
->     DECLARE new_res_id INT;
->     INSERT INTO Reservation (guest_id, check_in_date, check_out_date, booking_date, reservation_s
tatus)
->     VALUES (p_guest_id, p_checkin, p_checkout, CURDATE(), 'Booked');
->     SET new_res_id = LAST_INSERT_ID();
->     INSERT INTO Reserved_By (r_id, reservation_id)
->     VALUES (p_room_id, new_res_id);
->     UPDATE Room
->     SET r_status = 'Booked'
->     WHERE r_id = p_room_id;
-> END$$
Query OK, 0 rows affected (0.02 sec)

mysql> DELIMITER ;
mysql>
mysql> CALL sp_BookRoom(1, 1, '2025-11-15', '2025-11-18');
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM Reservation ORDER BY reservation_id DESC LIMIT 1;
+-----+-----+-----+-----+-----+-----+
| reservation_id | guest_id | check_in_date | check_out_date | booking_date | reservation_status |
+-----+-----+-----+-----+-----+-----+
|          99683 |         1 | 2025-11-15    | 2025-11-18    | 2025-11-13    | Booked              |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM Room WHERE r_id = 1;
+-----+-----+-----+-----+-----+-----+
| r_id | r_number | r_price | r_status | hotel_id | r_type_id |
+-----+-----+-----+-----+-----+-----+
|     1 | S101    | 2500.00 | Booked   |         1 |          1 |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```

mysql> DELIMITER $$
mysql> CREATE PROCEDURE sp_CancelReservation(IN p_reservation_id INT)
-> BEGIN
->     DECLARE v_room_id INT;
->     SELECT r_id INTO v_room_id
->     FROM Reserved_By
->     WHERE reservation_id = p_reservation_id;
->     UPDATE Reservation
->     SET reservation_status = 'Cancelled'
->     WHERE reservation_id = p_reservation_id;
->     UPDATE Room
->     SET r_status = 'Available'
->     WHERE r_id = v_room_id;
-> END$$
Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;
mysql>
mysql> CALL sp_CancelReservation(1);
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT reservation_id, reservation_status FROM Reservation WHERE reservation_id = 1;
+-----+-----+
| reservation_id | reservation_status |
+-----+-----+
|              1 | Cancelled          |
+-----+-----+
1 row in set (0.00 sec)

```

```

mysql> DELIMITER $$
mysql> CREATE FUNCTION fn_StayDuration(p_checkin DATE, p_checkout DATE)
-> RETURNS INT
-> DETERMINISTIC
-> BEGIN
->     RETURN DATEDIFF(p_checkout, p_checkin);
-> END$$
Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER ;
mysql>
mysql> SELECT fn_StayDuration('2025-11-10', '2025-11-12') AS Stay_Duration;
+-----+
| Stay_Duration |
+-----+
|              2 |
+-----+
1 row in set (0.00 sec)

```

## 8.2 TRIGGERS

```
mysql> DELIMITER $$
mysql> CREATE TRIGGER trg_UpdateRoomStatus
  -> AFTER INSERT ON Reserved_By
  -> FOR EACH ROW
  -> BEGIN
  ->     UPDATE Room
  ->     SET r_status = 'Booked'
  ->     WHERE r_id = NEW.r_id;
  -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql>
mysql> INSERT INTO Reserved_By (r_id, reservation_id) VALUES (2, 2);
Query OK, 1 row affected (0.01 sec)


mysql> SELECT r_id, r_status FROM Room WHERE r_id = 2;
+-----+-----+
| r_id | r_status |
+-----+-----+
|    2 | Booked  |
+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> DELIMITER $$
mysql> CREATE TRIGGER trg_PaymentLog
  -> AFTER INSERT ON Payment
  -> FOR EACH ROW
  -> BEGIN
  ->     INSERT INTO Payment_Log (payment_id, log_message)
  ->     VALUES (NEW.payment_id, CONCAT('Payment of ₹', NEW.payment_amount, ' received via ', NEW.payment_method));
  -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql>
mysql> INSERT INTO Payment (g_id, reservation_id, payment_amount, payment_method, payment_date)
  -> VALUES (1, 1, 6000, 'UPI', CURDATE());
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM Payment_Log ORDER BY log_id DESC LIMIT 5;
+-----+-----+-----+-----+
| log_id | payment_id | log_message                                     | log_time           |
+-----+-----+-----+-----+
|      1 |          4 | Payment of ?6000.00 received via UPI          | 2025-11-13 19:05:16 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

## 9. FRONT END DEVELOPMENT



### Hotel Management

[Guest](#)[Admin](#)

Full Name \*

Email Address \*

Phone Number

City

State

sanjana

Logout



### Welcome, sanjana!

[Book Room](#)[My Reservations](#)[Make Payment](#)

#### Book a Room

Select Hotel \*

Marriott Bangalore

Select Room \*

D201 - Deluxe (₹4200.00/night)

Check-in Date \*

14 - 11 - 2025


Check-out Date \*

18 - 11 - 2025

Book Room

sanjana

Logout

 **Welcome, sanjana!**

[Book Room](#) [My Reservations](#) [Make Payment](#)

**Make Payment**


Select Reservation to Pay \*

Reservation 1012 - Room D201 (2025-11-14 to 2025-11-18) ▾


Total Amount to Pay

₹ 16,800

Payment Method \*

 Net Banking ▾

Process Payment

**Hotel Management**

[Guest](#) [Admin](#)

Username

admin\_oberoi

Password

\*\*\*\*\*

Admin Login

ADMIN

admin\_oberoi

Logout

Admin Dashboard

All ReservationsRoom StatusPayment Logs

All Reservations

Res ID	Guest Name	Email	Check-In	Check-Out	Room	Status
1009	sanjana	sanjana@gmail.com	2025-11-10	2025-11-14	F402	Booked
1005	risu	risu@abc.com	2025-11-07	2025-11-09	S101	Booked

Total Reservations: 2

admin\_oberoi

Logout

All ReservationsRoom StatusPayment Logs

Room Status

Room Number	Type	Price (₹)	Status
S101	Single Room	2800.00	Booked
S102	Single Room	2800.00	Available
S103	Single Room	2800.00	Available
S104	Single Room	2800.00	Available
D201	Deluxe	4500.00	Available
D202	Deluxe	4500.00	Available
D203	Deluxe	4500.00	Available
D204	Deluxe	4500.00	Available
T301	Twin Bed	3800.00	Available

ADMIN

admin\_oberoi

Logout



## Admin Dashboard

All Reservations

Room Status

Payment Logs

### Payment Logs

Log ID	Payment ID	Guest Name	Room	Message	Timestamp
4	5	sanjana	SU403	Payment of ₹25500.00 via Card	2025-11-13 11:44:04
5	5	sanjana	SU403	Payment of ₹25500.0 received via Card	2025-11-13 11:44:04
3	4	risu	F401	Payment of ₹24800.0 received via UPI	2025-11-07 14:18:54
2	3	risu	S101	Payment of ₹5600.0 received via Card	2025-11-07 08:41:47
1	2	sanjana	D201	Payment of ₹7000.0 received via Net Banking	2025-11-06 22:27:21

Total Payment Logs: 5

# REFERENCES

- [1] *Silberschatz, Abraham; Korth, Henry F.; Sudarshan, S.*, **“Database System Concepts,”** 7th Edition, McGraw-Hill Education, 2020.
- [2] *MySQL Documentation*, **“MySQL 8.0 Reference Manual,”** Oracle Corporation. Available at: <https://dev.mysql.com/doc/>
- [3] *Flask Documentation*, **“Flask Web Framework (Python),”** Pallets Projects. Available at: <https://flask.palletsprojects.com/>
- [4] *W3Schools*, **“HTML5 Tutorial – Building Frontend Templates,”** Available at: <https://www.w3schools.com/html/>
- [5] *GeeksforGeeks*, **“Flask – Introduction, Routing, and Jinja Templates,”** Available at: <https://www.geeksforgeeks.org/flask-tutorial/>
- [6] *MDN Web Docs (Mozilla)*, **“HTML & CSS Documentation for Frontend Design,”** Available at: <https://developer.mozilla.org/>

# APPENDIX A DEFINITIONS, ACRONYMS AND ABBREVIATIONS

## A.1 Definitions

<u>Term</u>	<u>Definition</u>
Database	An organized collection of structured information stored electronically.
Reservation	A booking made by a guest for a hotel room for a specific date range.
Room Status	Indicates whether a hotel room is <i>Available</i> , <i>Booked</i> , <i>Occupied</i> , or <i>Under Maintenance</i> .
Trigger	A database object that automatically executes when an event occurs (INSERT/UPDATE/DELETE).
Stored Procedure	A reusable block of SQL statements stored in the database and executed when called.
Foreign Key	A constraint used to link two database tables based on a related column.
Frontend	User interface part of the application that users interact with (HTML/CSS/Jinja templates).
Backend	Server-side logic that processes data, executes queries, and manages application flow (Flask).

## A.2 Acronyms

<u>Acronym</u>	<u>Meaning</u>
DBMS	Database Management System
SQL	Structured Query Language
UI	User Interface
CRUD	Create, Read, Update, Delete
HTML	HyperText Markup Language
CSS	Cascading Style Sheets

API	Application Programming Interface
IDE	Integrated Development Environment

### **A.3 Abbreviations Used in the Database**

<u>Abbreviation</u>	<u>Meaning</u>
g_id	Guest ID
h_id	Hotel ID
r_id	Room ID
room_type_id	Room Type ID
reservation_id	Reservation ID
a_user_id	Administrator User ID
payment_id	Payment ID
res_id	Reservation ID (in Administrator table)

## **GITHUB REPO LINK**

<https://github.com/RisuKumari11/hotel-reservation-system>