

Hotel Room Booking System

By- Samiksha.V.Salunkhe

Home

About

Contact



Welcome to Larana Hotel

The Hotel Room Booking System is designed to manage guest information, room availability, bookings, payments, and services offered by a hotel. This project simulates a real-world hotel management database system, using SQL to store, retrieve, and analyze data efficiently.

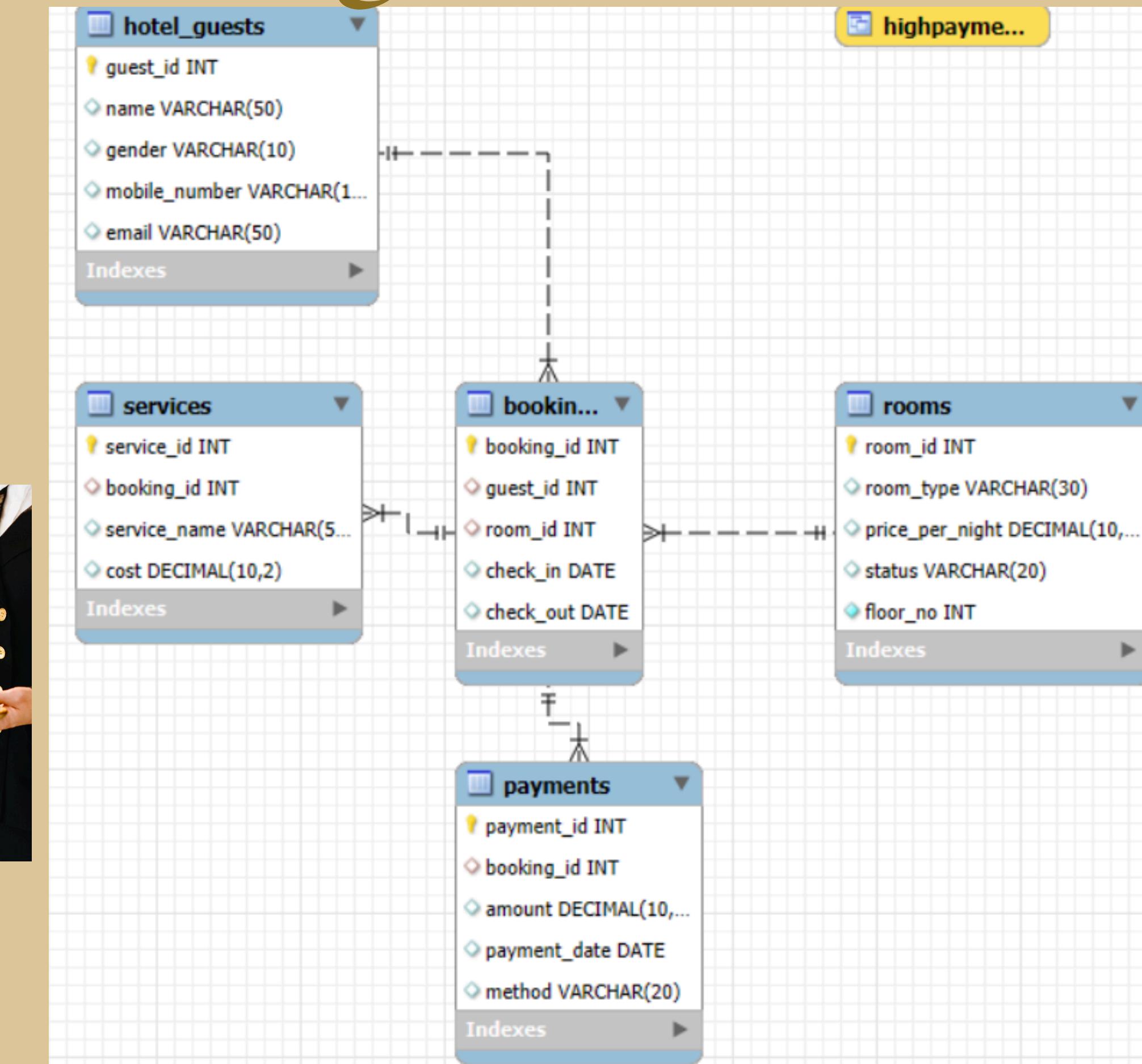
The system helps hotel staff:

- Track room occupancy and availability.
- Handle guest check-in and check-out.
- Manage service records and payments.
- Generate reports like revenue by room type or peak booking periods.





ER Diagram

[About](#)

Databases :

CREATE DATABASE HotelDB;

USE HotelDB;

Show databases;

Database
college
company
hoteldb
information_schema
joins_sql
mysql

About

Contact

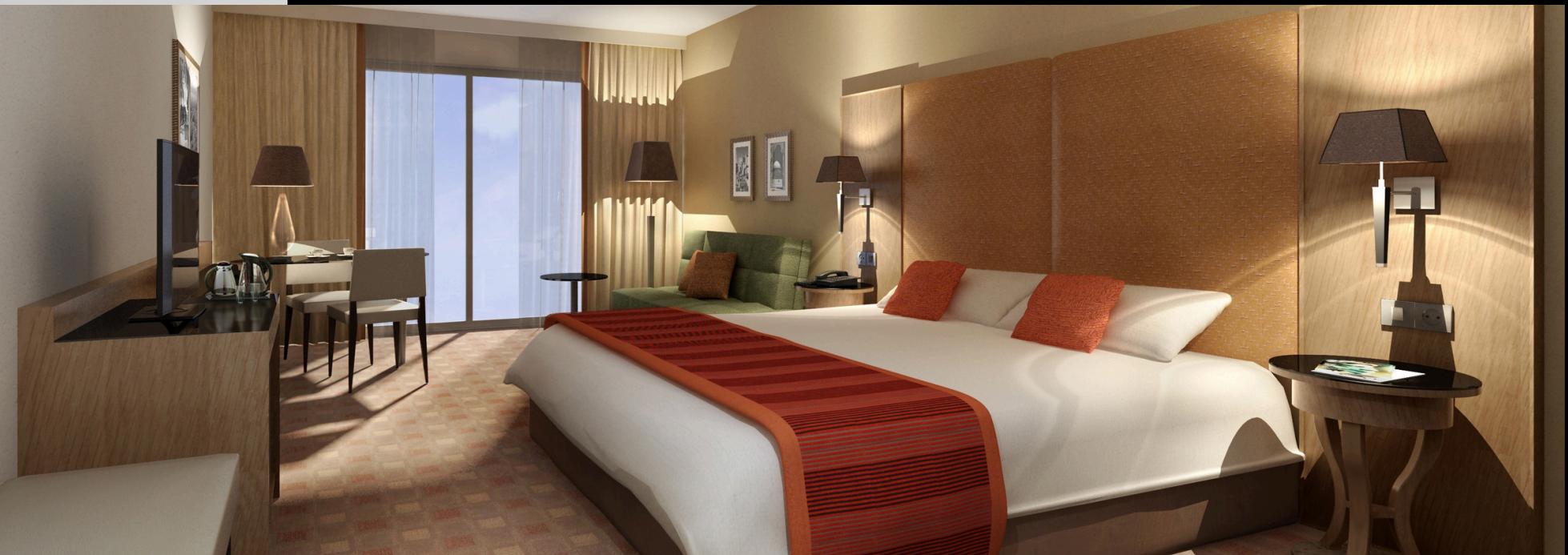


Tables in Hospital_data

Database :

show tables;

Tables_in_hoteldb
bookings
guests
payments
rooms
services



Data Definition Language (DDL):

1. Creating Tables:

A) Bookings

Field	Type	Null	Key	Default	Extra
booking_id	int	NO	PRI	NULL	
guest_id	int	YES	MUL	NULL	
room_id	int	YES	MUL	NULL	
check_in	date	YES		NULL	
check_out	date	YES		NULL	

CREATE TABLE Bookings (

booking_id INT PRIMARY KEY,
guest_id INT,
room_id INT,
check_in DATE,
check_out DATE,

FOREIGN KEY (guest_id) REFERENCES

Guests(guest_id),

FOREIGN KEY (room_id) REFERENCES

Rooms(room_id)

);



B) Rooms

Field	Type	Null	Key	Default
room_id	int	NO	PRI	NULL
room_type	varchar(30)	YES		NULL
price_per_night	decimal(10,2)	YES		NULL
status	varchar(20)	YES		NULL

```
CREATE TABLE Rooms (
    room_id INT PRIMARY KEY,
    room_type VARCHAR(30),
    price_per_night DECIMAL(10,2),
    status VARCHAR(20) -- e.g., 'Available',
    'Booked'
);
```

C) Services

	Field	Type	Null	Key	Default	Extra
	service_id	int	NO	PRI	NULL	
	booking_id	int	YES	MUL	NULL	
	service_name	varchar(50)	YES		NULL	
	cost	decimal(10,2)	YES		NULL	

```
CREATE TABLE Services (
    service_id INT PRIMARY KEY,
    booking_id INT,
    service_name VARCHAR(50),
    cost DECIMAL(10,2),
    FOREIGN KEY (booking_id) REFERENCES
        Bookings(booking_id)
);
```

D) Guests

	Field	Type	Null	Key	Default	Extra
▶	guest_id	int	NO	PRI	HULL	
	name	varchar(50)	YES		HULL	
	gender	varchar(10)	YES		HULL	
	phone	varchar(15)	YES		HULL	
	email	varchar(50)	YES		HULL	

E) Payments

	Field	Type	Null	Key	Default	Extra
▶	payment_id	int	NO	PRI	HULL	
	booking_id	int	YES	MUL	HULL	
	amount	decimal(10,2)	YES		HULL	
	payment_date	date	YES		HULL	
	method	varchar(20)	YES		HULL	

CREATE TABLE Payments (

payment_id INT PRIMARY KEY,

booking_id INT,

amount DECIMAL(10,2),

payment_date DATE,

method VARCHAR(20), -- e.g., 'Card', 'Cash'

FOREIGN KEY (booking_id) REFERENCES

Bookings(booking_id)

);

CREATE TABLE Guests (

guest_id INT PRIMARY KEY,

name VARCHAR(50),

gender VARCHAR(10),

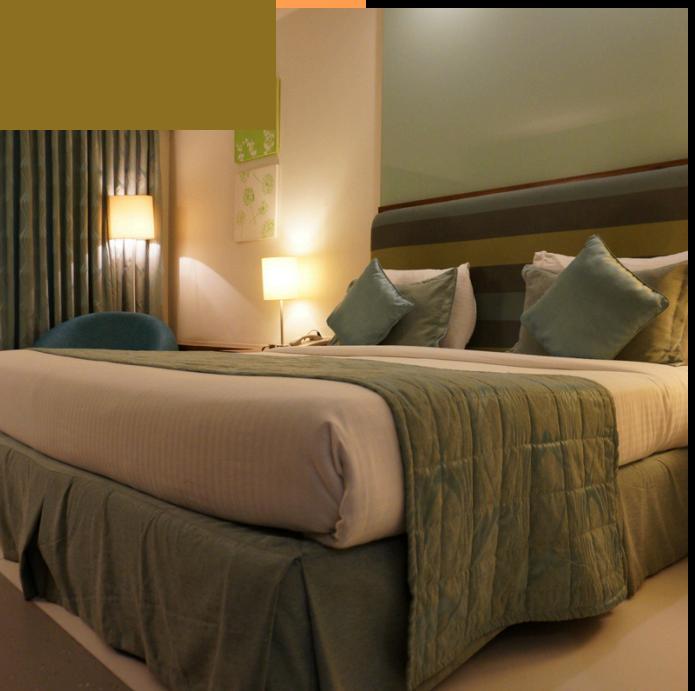
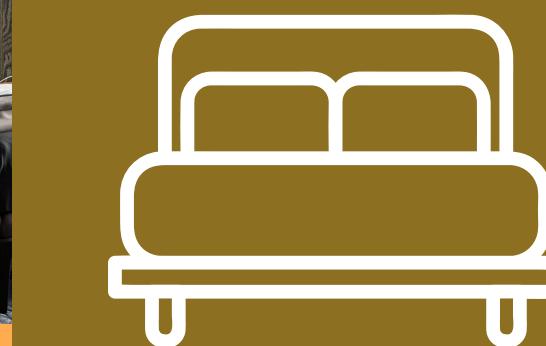
phone VARCHAR(15),

email VARCHAR(50)

);

About

Contact



2. Alter Table

- Alter Table : Add column

ALTER TABLE Rooms ADD floor_no INT NOT NULL DEFAULT 0;

Field	Type	Null	Key	Default	Extra
room_id	int	NO	PRI	HULL	
room_type	varchar(30)	YES		HULL	
price_per_night	decimal(10,2)	YES		HULL	
status	varchar(20)	YES		HULL	
floor_no	int	NO		0	

- Alter Table : Modify Column

alter table payments modify method varchar(60);

Field	Type	Null	Key	Default	Extra
payment_id	int	NO	PRI	HULL	
booking_id	int	YES	MUL	HULL	
amount	decimal(10,2)	YES		HULL	
payment_date	date	YES		HULL	
method	varchar(60)	YES		HULL	

- Alter Table : Rename Column

ALTER TABLE Guests RENAME COLUMN phone TO mobile_number;

	Field	Type	Null	Key	Default	Extra
▶	payment_id	int	NO	PRI	HULL	
	booking_id	int	YES	MUL	HULL	
	amount	decimal(10,2)	YES		HULL	
	payment_date	date	YES		HULL	

- Alter Table : Rename Table

ALTER TABLE Guests RENAME TO Hotel_Guests;

	Field	Type	Null	Key	Default	Extra
▶	guest_id	int	NO	PRI	HULL	
	name	varchar(50)	YES		HULL	
	gender	varchar(10)	YES		HULL	
	mobile_number	varchar(15)	YES		HULL	
	email	varchar(50)	YES		HULL	

- Alter Table : Drop column

ALTER TABLE Guests RENAME COLUMN phone TO mobile_number;

	Field	Type	Null	Key	Default	Extra
▶	guest_id	int	NO	PRI	HULL	
	name	varchar(50)	YES		HULL	
	gender	varchar(10)	YES		HULL	
	mobile_number	varchar(15)	YES		HULL	
	email	varchar(50)	YES		HULL	



3.Truncate table :

Truncate table Services;

service_id	booking_id	service_name	cost
HULL	HULL	HULL	HULL

4. Drop table :

drop table Services;

Data Manipulation Language (DML):

1.Insert into Tables :

```
INSERT INTO hotel_Guests (guest_id, name, gender, mobile_number, email)
VALUES (1, 'Ravi Sharma', 'Male', '9876543210', 'ravi@gmail.com');
select* from hotel_guests;
```

	guest_id	name	gender	mobile_number	email
▶	1	Ravi Sharma	Male	9876543210	ravi@gmail.com
	2	Pooja Mehta	Female	8765432109	pooja@gmail.com
	3	Amit Verma	Male	9988776655	amitv@yahoo.com
	4	Sneha Kulkarni	Female	8899776655	sneha.kulkarni@hotmail.com
	5	Rahul Singh	Male	9012345678	rahulsingh@outlook.com
	6	Neha Jain	Female	9123456780	nehajain@gmail.com
	7	Aditya Desai	Male	9345678901	aditya.desai@gmail.com
	8	Kavita Rao	Female	9456781234	kavita.rao@yahoo.com
	9	Manoj Kumar	Male	95 9456781234	manojkumar@gmail.com
	10	Deepika Sharma	Female	9678123456	deepika.sharma@gmail.com
	HULL	HULL	HULL	HULL	HULL

2.Update into Tables :

Q. Update the phone and email of the guest with ID 2

```
UPDATE hotel_guests SET mobile_number = '9871234560',  
email = 'pooja123@gmail.com' WHERE guest_id = 2;
```

[Home](#)[About](#)[Contact](#)

guest_id	name	gender	mobile_number	email
1	Ravi Sharma	Male	9876543210	ravi@gmail.com
2	Pooja Mehta	Female	9871234560	pooja123@gmail.com
3	Amit Verma	Male	9988776655	amity@yahoo.com
4	Sneha Kulkarni	Female	8899776655	sneha.kulkarni@hotmail.com
5	Rahul Singh	Male	9012345678	rahulsingh@outlook.com
6	Neha Jain	Female	9123456780	nehajain@gmail.com
7	Aditya Desai	Male	9345678901	aditya.desai@gmail.com
8	Kavita Rao	Female	9456781234	kavita.rao@yahoo.com
9	Manoj Kumar	Male	9567812345	manojkumar@gmail.com
10	Deepika Sharma	Female	9678123456	deepika.sharma@gmail.com
NULL	NULL	NULL	NULL	NULL

3.Delete from Tables

Q Delete record having guest_id=10

```
delete from hotel_guests where guest_id = 10;
```

guest_id	name	gender	mobile_number	email
1	Ravi Sharma	Male	9876543210	ravi@gmail.com
2	Pooja Mehta	Female	9871234560	pooja123@gmail.com
3	Amit Verma	Male	9988776655	amity@yahoo.com
4	Sneha Kulkarni	Female	8899776655	sneha.kulkarni@hotmail.com
5	Rahul Singh	Male	9012345678	rahulsingh@outlook.com
6	Neha Jain	Female	9123456780	nehajain@gmail.com
7	Aditya Desai	Male	9345678901	aditya.desai@gmail.com
8	Kavita Rao	Female	9456781234	kavita.rao@yahoo.com
9	Manoj Kumar	Male	9567812345	manojkumar@gmail.com
NULL	NULL	NULL	NULL	NULL



Data Query Language (DDL):

1. Select Query:

a) Select Query for entire data

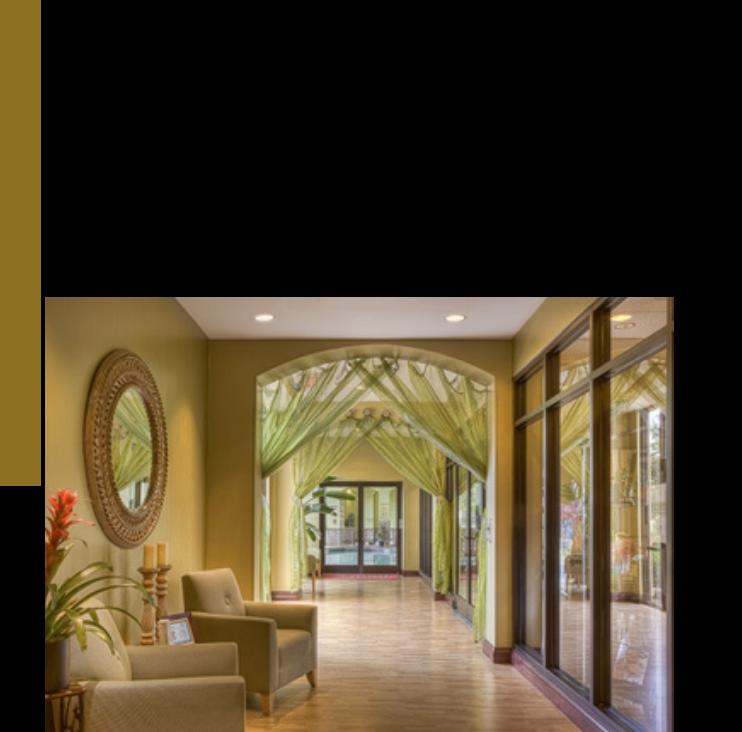
```
select*from hotel_guests;
```

guest_id	name	gender	mobile_number	email
1	Ravi Sharma	Male	9876543210	ravi@gmail.com
2	Pooja Mehta	Female	9871234560	pooja123@gmail.com
3	Amit Verma	Male	9988776655	amitv@yahoo.com
4	Sneha Kulkarni	Female	8899776655	sneha.kulkarni@hotmail.com
5	Rahul Singh	Male	9012345678	rahulsingh@outlook.com
6	Neha Jain	Female	9123456780	nehajain@gmail.com
7	Aditya Desai	Male	9345678901	aditya.desai@gmail.com
8	Kavita Rao	Female	9456781234	kavita.rao@yahoo.com
9	Manoj Kumar	Male	9567812345	manojkumar@gmail.com
NONE	NONE	NONE	NONE	NONE

b) Select specific data i.e name and gender

name	gender
Ravi Sharma	Male
Pooja Mehta	Female
Amit Verma	Male
Sneha Kulkarni	Female
Rahul Singh	Male
Neha Jain	Female
Aditya Desai	Male
Kavita Rao	Female
Manoj Kumar	Male

Select name,gender
from hotel_guests;



c) Select query with changing column name

```
select name as guest_name from  
hotel_guests;
```

guest_name
Ravi Sharma
Pooja Mehta
Amit Verma
Sneha Kulkarni
Rahul Singh
Neha Jain
Aditya Desai
Kavita Rao
Manoj Kumar

2. Order By

a) List of Services in ascending order by cost.

select * from services order by cost;

	service_id	booking_id	service_name	cost
▶	301	201	Room Cleaning	500.00
	302	202	Laundry	800.00
	303	203	Food Delivery	1200.00
●	NULl	NULl	NULl	NULl

b) List of Services in descending order by cost

select * from services order by cost desc;

	service_id	booking_id	service_name	cost
▶	303	203	Food Delivery	1200.00
	302	202	Laundry	800.00
	301	201	Room Cleaning	500.00
●	NULl	NULl	NULl	NULl

3. Limit Query

Show top 2 highest-paying bookings

SELECT * FROM Payments ORDER BY amount DESC LIMIT 2;

	payment_id	booking_id	amount	payment_date	method
▶	503	203	20000.00	2025-07-03	UPI
	501	201	14000.00	2025-07-01	Card
●	NULl	NULl	NULl	NULl	NULl

4. Distinct Query

Display Unique Gender from Guests.

Select distinct gender from hotel_guests;;

	gender
▶	Male
●	Female



5. Where Clause

1) with Comparison Operator

Q. Find bookings with checkout after July 5, 2025

SELECT * FROM Bookings WHERE check_out > '2025-07-05';

	booking_id	guest_id	room_id	check_in	check_out
▶	202	2	102	2025-07-02	2025-07-06
▶	203	3	103	2025-07-03	2025-07-07
●	NULL	NULL	NULL	NULL	NULL

2) Logical Operator

- Using AND Operator

Q. Payments made using 'Card' and amount > ₹10000

SELECT * FROM Payments WHERE method = 'Card' AND amount > 10000;

	payment_id	booking_id	amount	payment_date	method
▶	501	201	14000.00	2025-07-01	Card
●	NULL	NULL	NULL	NULL	NULL

- Using AND /OR Operator

Q. Rooms that are 'Available' AND Deluxe OR any Suite

SELECT * FROM Rooms WHERE (status = 'Available' AND room_type = 'Deluxe') OR room_type = 'Suite';

	room_id	room_type	price_per_night	status	floor_no
▶	101	Deluxe	3500.00	Available	0
▶	103	Suite	5000.00	Available	0
●	NULL	NULL	NULL	NULL	NULL

- Using NOT Operator

Q. Services not costing more than ₹1000

SELECT * FROM Services WHERE NOT cost > 1000;

	service_id	booking_id	service_name	cost
▶	301	201	Room Cleaning	500.00
	302	202	Laundry	800.00
	HULL	HULL	HULL	HULL

- Using NOT NULL

Q. Payments with payment method entered

SELECT * FROM Payments WHERE method IS NOT NULL;

	payment_id	booking_id	amount	payment_date	method
▶	501	201	14000.00	2025-07-01	Card
	502	202	10000.00	2025-07-02	Cash
	503	203	20000.00	2025-07-03	UPI
	HULL	HULL	HULL	HULL	HULL

- Using Between

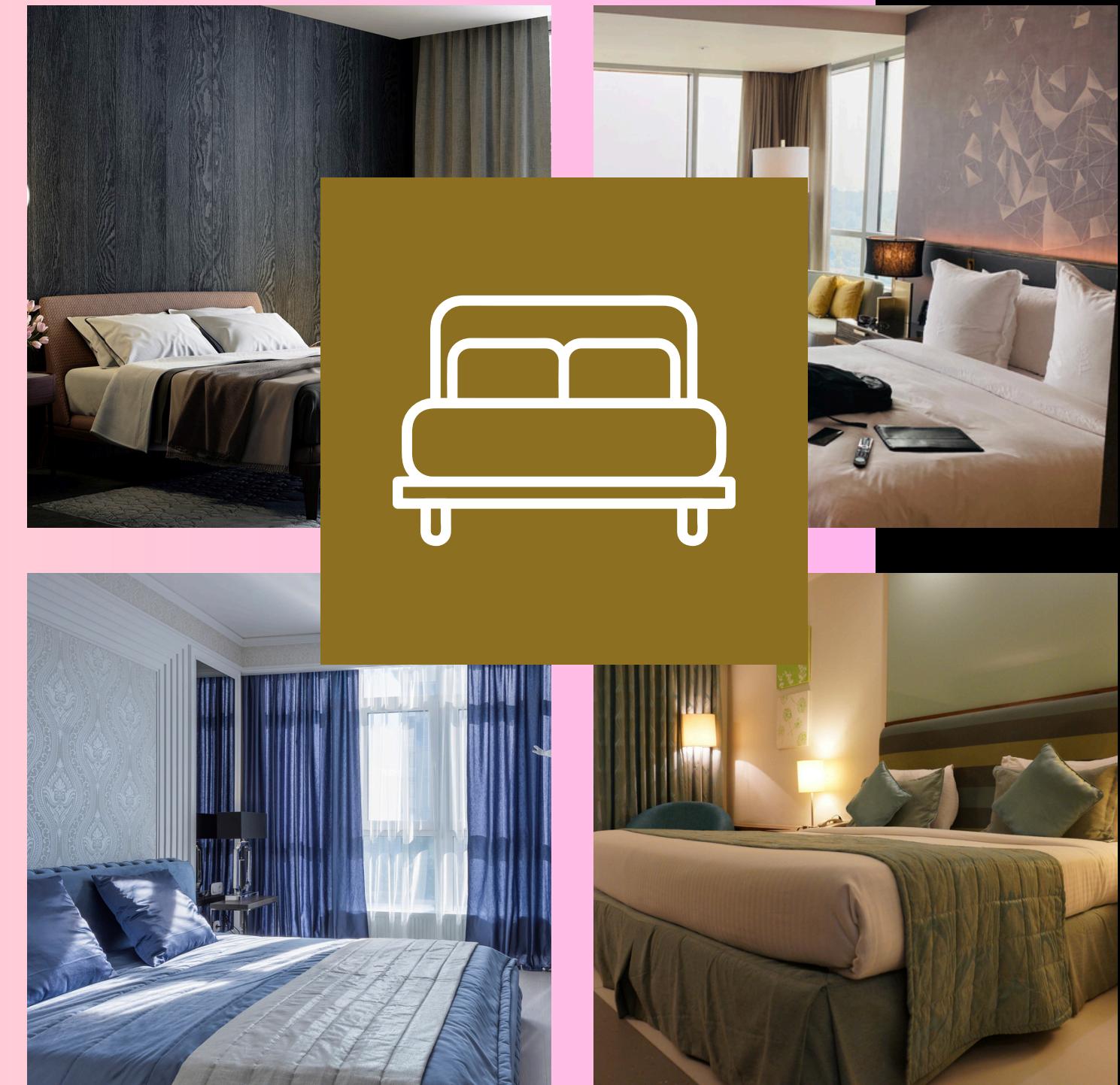
Q. Guests with IDs between 4 and 7

SELECT * FROM hotel_Guests WHERE guest_id BETWEEN 4 AND 7;

	guest_id	name	gender	mobile_number	email
▶	4	Sneha Kulkarni	Female	8899776655	sneha.kulkarni@hotmail.com
	5	Rahul Singh	Male	9012345678	rahulsingh@outlook.com
	6	Neha Jain	Female	9123456780	nehajain@gmail.com
	7	Aditya Desai	Male	9345678901	aditya.desai@gmail.com
	HULL	HULL	HULL	HULL	HULL

About

Contact



• Using IN

Q. Payments made by 'Card' or 'UPI'

```
SELECT * FROM Payments WHERE method IN ('Card', 'UPI');
```

	payment_id	booking_id	amount	payment_date	method
▶	501	201	14000.00	2025-07-01	Card
	503	203	20000.00	2025-07-03	UPI
◀	HULL	HULL	HULL	HULL	HULL

About

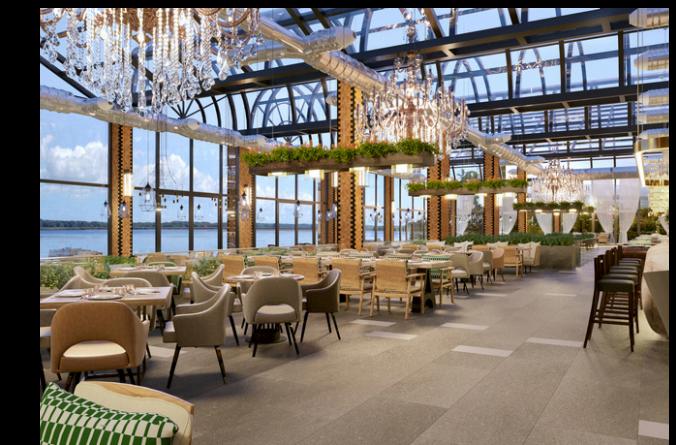


• Using ALL

Q. Rooms with prices higher than all Standard rooms

```
SELECT * FROM Rooms WHERE price_per_night > ALL (SELECT
    price_per_night FROM Rooms WHERE room_type = 'Standard');
```

	room_id	room_type	price_per_night	status	floor_no
▶	101	Deluxe	3500.00	Available	0
	103	Suite	5000.00	Available	0
◀	HULL	HULL	HULL	HULL	HULL



• Using ANY

Q. Find rooms more expensive than any Standard room

```
SELECT * FROM Rooms WHERE price_per_night > ANY (SELECT
    price_per_night FROM Rooms WHERE room_type = 'Standard');
```

	room_id	room_type	price_per_night	status	floor_no
▶	101	Deluxe	3500.00	Available	0
	103	Suite	5000.00	Available	0
◀	HULL	HULL	HULL	HULL	HULL





6. Aggregate Functions

- **Count**

Q. Find the total number of guests

Select count(*) as total_number_guests from Hotel_guests;

	total_number_guests
▶	9

- **Average Function with round function**

Q. Average service cost (1 decimal place)

SELECT ROUND(AVG(cost),1) AS avg_service_cost FROM Services;

	avg_service_cost
▶	833.3

- **SUM**

Q. Total payment received

SELECT SUM(amount) AS total_revenue FROM Payments;

	total_revenue
▶	44000.00

- **MIN,MAX**

Q. Find highest and lowest room prices

highest_price	lowest_price
5000.00	2500.00

**SELECT MAX(price_per_night) AS highest_price,
MIN(price_per_night) AS lowest_price
FROM Rooms;**

About

7. Group By Clause

Q. Count of guests by gender

```
SELECT gender, COUNT(guest_id) AS number_of_guests FROM Guests GROUP BY gender;
```

	gender	number_of_guests
▶	Male	5
	Female	5

Q. Average room price by room type

```
SELECT room_type, ROUND(AVG(price_per_night), 2) AS avg_price FROM Rooms GROUP BY room_type;
```

	room_type	avg_price
▶	Deluxe	3500.00
	Standard	2500.00
	Suite	5000.00

8. Like Operator

Q. Guests whose name starts with 'S'

```
SELECT * FROM Guests WHERE name LIKE 'S%';
```

	guest_id	name	gender	phone	email
▶	4	Sneha Kulkarni	Female	8899776655	sneha.kulkarni@hotmail.com
◀	HULL	HULL	HULL	HULL	HULL

9.Union Operator

Q.Show all payment methods and service names (as a single list)

Home

About

Contact

```
SELECT method AS description FROM Payments UNION  
SELECT service_name AS description FROM Services;
```

	description
▶	Card
	Cash
	UPI
	Room Cleaning
	Laundry
	Food Delivery

10. Having Clause

Q. Guests who booked 1st time

```
SELECT guest_id, COUNT(booking_id) AS booking_count FROM  
Bookings GROUP BY guest_id HAVING COUNT(booking_id) > 0;
```

	guest_id	total_bookings
▶	1	1
	2	1
	3	1



11. Joins

Q. INNER JOIN: Booking with Guest Info

```
SELECT b.booking_id, g.name, b.check_in, b.check_out  
FROM Bookings b INNER JOIN Guests g ON b.guest_id = g.guest_id;
```



Q. LEFT JOIN: All Guests and Their Bookings

```
SELECT g.name, b.booking_id, b.check_in FROM Guests g  
LEFT JOIN Bookings b ON g.guest_id = b.guest_id;
```



	name	booking_id	check_in
▶	Ravi Sharma	201	2025-07-01
	Pooja Mehta	202	2025-07-02
	Amit Verma	203	2025-07-03
	Sneha Kulkarni	NULL	NULL
	Rahul Singh	NULL	NULL
	Neha Jain	NULL	NULL
	Aditya Desai	NULL	NULL
	Kavita Rao	NULL	NULL
	Manoj Kumar	NULL	NULL
	Deepika Sharma	NULL	NULL

12. SubQueries

1. Single-Row Subquery

Q. Get payments higher than the average payment amount.

`SELECT * FROM Payments WHERE amount > (SELECT AVG(amount) FROM Payments);`

	payment_id	booking_id	amount	payment_date	method
▶	503	203	20000.00	2025-07-03	UPI
◀	NUL	NUL	NUL	NUL	NUL

2. Multiple- Column Subquery

Q. Display the payment ID, method, and amount for the highest payment made under each payment method

`SELECT payment_id, method, amount FROM Payments WHERE (method, amount)
IN (SELECT method, MAX(amount) FROM Payments GROUP BY method);`

	payment_id	method	amount
▶	501	Card	14000.00
	502	Cash	10000.00
	503	UPI	20000.00
◀	NUL	NUL	NUL

3. Multiple- Row Subquery

Q. Guests who had made booking.

`SELECT * FROM Guests WHERE guest_id IN (SELECT guest_id FROM Bookings);`

	guest_id	name	gender	phone	email
▶	1	Ravi Sharma	Male	9876543210	ravi@gmail.com
	2	Pooja Mehta	Female	8765432109	pooja@gmail.com
	3	Amit Verma	Male	9988776655	amity@yahoo.com
◀	NUL	NUL	NUL	NUL	NUL



Q. View for high-value payments

```
CREATE VIEW HighPayments AS SELECT * FROM Payments  
WHERE amount > 10000;  
SELECT * FROM HighPayments;
```

	payment_id	booking_id	amount	payment_date	method
▶	501	201	14000.00	2025-07-01	Card
	503	203	20000.00	2025-07-03	UPI

Thank
You
FOR YOUR ATTENTION