

HOTEL_RESERVATION_SYSTEM Documentation

Database Initialization

- **Database Created:** HOTEL_RESERVATION_SYSTEM

SYNTAX: `CREATE DATABASE HOTEL_RESERVATION_SYSTEM;`

`USE HOTEL_RESERVATION_SYSTEM;`

Table Creation

- **Table Created:** hotel_reservations

SYNTAX: `CREATE TABLE hotel_reservations (reservation_id INT PRIMARY KEY, guest_name VARCHAR (100), room_number INT, check_in_date DATE, check_out_date DATE, number_of_guests INT);`

Adding a new column to table using alter:

SYNTAX: `ALTER TABLE hotel_reservations ADD payment_status VARCHAR(20);`

Data Manipulation

◆ Inserted Records

10 initial reservations were added with varied payment statuses: Paid, Pending, and Cancelled.

SYNTAX: `INSERT INTO hotel_reservations (reservation_id, guest_name, room_number, check_in_date, check_out_date, number_of_guests, payment_status)`

VALUES

(101, 'Rajeswari', 408, '2025-08-22', '2025-08-26', 2, 'Paid'),
(102, 'Bavagnya', 115, '2025-08-20', '2025-08-22', 1, 'Pending'),
(103, 'Sesi', 312, '2025-08-21', '2025-08-24', 4, 'Paid'),
(104, 'Dharani', 207, '2025-08-23', '2025-08-27', 2, 'Cancelled'),
(105, 'Abinaya', 101, '2025-08-19', '2025-08-21', 1, 'Paid'),
(106, 'Ram', 410, '2025-08-22', '2025-08-26', 2, 'Paid'),
(107, 'Nani', 235, '2025-08-20', '2025-08-22', 1, 'Pending'),
(108, 'Sam', 352, '2025-08-21', '2025-08-24', 4, 'Paid'),
(109, 'Durga', 307, '2025-08-23', '2025-08-27', 2, 'Cancelled'),
(110, 'Varshini', 201, '2025-08-23', '2025-08-21', 1, 'Paid');

◆ Updates

- Changed payment_status of reservation ID 103 to Pending.

SYNTAX: `UPDATE hotel_reservations SET payment_status = 'Pending' WHERE reservation_id = 103;`

◆ Deletions

- Removed reservation with ID 107.

SYNTAX: `DELETE FROM hotel_reservations WHERE reservation_id = 107;`

Queries and Analysis

Viewing all records in table

- `SELECT * FROM hotel_reservations` — View all records.

Filtered by Payment Status

- `SELECT * WHERE payment_status = 'Paid'`

Aggregate Functions :

Set of operations that performs calculations on set of values and returns one value.

There are different types of aggregate functions : `Count()`,`max()`,`min()`,`sum()`,`avg()`.

EXAMPLE QUERIES: `SELECT COUNT(*) AS total_reservations FROM hotel_reservations;`

`SELECT SUM(number_of_guests) AS total_guests FROM hotel_reservations;`

`SELECT AVG(number_of_guests) AS avg_guests_per_booking FROM hotel_reservations;`

`SELECT MIN(check_in_date) AS earliest_check_in, MAX(check_in_date) AS latest_check_in`

`FROM hotel_reservations;`

Grouping and Filtering (HAVING CLAUSE)

- Grouped bookings by `payment_status`

`SELECT payment_status, COUNT(*) AS total_bookings FROM hotel_reservations GROUP BY payment_status;`

- Used HAVING to filter check-in dates with average guests > 3

`SELECT check_in_date, AVG(number_of_guests) FROM hotel_reservations GROUP BY check_in_date HAVING AVG(number_of_guests) > 3;`

Pattern Matching (using LIKE keyword)

- Guests with names starting with 'D': `LIKE 'D%'`

`SELECT * FROM hotel_reservations WHERE guest_name LIKE 'D%';`

Subqueries

- Retrieved guest names with Paid status using nested SELECT

`SELECT guest_name FROM hotel_reservations WHERE reservation_id IN (SELECT reservation_id FROM hotel_reservations WHERE payment_status = 'Paid');`

Stored Procedure: Allows dynamic insertion

Allows dynamic insertion of new reservations using parameters.

DELIMITER //

```
CREATE PROCEDURE AddHotelReservation (IN p_reservation_id INT, IN p_guest_name VARCHAR(100),
IN p_room_number INT, IN p_check_in_date DATE, IN p_check_out_date DATE, IN p_number_of_guests INT)
BEGIN
    INSERT INTO hotel_reservations (reservation_id, guest_name, room_number, check_in_date, check_out_date,
    number_of_guests)
    VALUES (p_reservation_id, p_guest_name, p_room_number, p_check_in_date, p_check_out_date,
    p_number_of_guests);
END;
CALL AddHotelReservation(111, 'NANI', 204, '2025-08-21', '2025-08-24', 2);
```

Trigger: Set of action that are automatically executed in response to an event such insert, update, delete operations on a table

```
DELIMITER //
CREATE TRIGGER set_default_payment_status
BEFORE INSERT ON hotel_reservations
FOR EACH ROW
BEGIN
IF NEW.payment_status IS NULL THEN
    SET NEW.payment_status = 'Pending';
END IF;
END;
//
DELIMITER ;
```