**HOTEL\_RESERVATION\_SYSTEM Documentation**

**Database Initialization**

* **Database Created:** HOTEL\_RESERVATION\_SYSTEM

**Table Creation**

* **Table Created:** hotel\_reservations

**Table Schema**

| **Column Name** | **Data Type** | **Description** |
| --- | --- | --- |
| reservation\_id | INT | Primary key |
| guest\_name | VARCHAR (100) | Name of the guest |
| room\_number | INT | Room number assigned |
| check\_in\_date | DATE | Check-in date |
| check\_out\_date | DATE | Check-out date |
| number\_of\_guests | INT | Number of guests |
| payment\_status | VARCHAR (20) | Status of payment (added later using alter add) |

**Data Manipulation**

**🔹 Inserted Records**

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

**🔹 Updates**

* Changed payment\_status of reservation ID 103 to Pending.

**🔹 Deletions**

* Removed reservation with 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**

| **Query Purpose** | **SQL Function Used** |
| --- | --- |
| Total reservations | COUNT (\*) |
| Total guests | SUM (number\_of\_guests) |
| Average guests per booking | AVG (number\_of\_guests) |
| Earliest & Latest check-in | MIN, MAX |

**Grouping and Filtering**

* Grouped bookings by payment\_status
* Used HAVING to filter check-in dates with average guests > 3

**Pattern Matching**

* Guests with names starting with 'D': LIKE 'D%'

**Subqueries**

* Retrieved guest names with Paid status using nested SELECT

**Stored Procedure: AddHotelReservation**

Allows dynamic insertion of new reservations using parameters.

**Trigger: prevent\_double\_booking**

Prevents overlapping bookings for the same room by checking date conflicts before insertion.