**Hotel Management System**

**Abstract:**

**The Hotel Management System (HMS) is a streamlined software solution designed for effortless hotel operations. It seamlessly manages room reservations, guest details, staff assignments, and financial transactions, ensuring a seamless guest experience. From intuitive room allocation to simplified check-ins and check-outs, the system optimizes guest services. It also assists staff management, tracks maintenance, handles event bookings, and gathers guest feedback for continuous improvement. With its user-friendly interface and adaptable features, the HMS empowers hotels to enhance efficiency, deliver personalized service, and elevate guest satisfaction.**

**Aim:**

**The aim of the Hotel Management System (HMS) is to create an efficient, user-friendly, and automated platform that simplifies hotel operations. By integrating reservation management, guest services, staff administration, and financial transactions, the system aims to enhance the overall guest experience, optimize staff productivity, and improve the hotel's operational efficiency.**

**Introduction:**

**In the modern hospitality industry, managing diverse hotel operations manually is not only time-consuming but also prone to errors. The Hotel Management System (HMS) is introduced as a robust software solution designed to revolutionize hotel management practices. By automating tasks such as room bookings, guest check-ins, staff scheduling, and financial transactions, the HMS not only streamlines operations but also ensures a seamless and personalized experience for guests. This introduction sets the stage for a more efficient and guest-focused hotel management approach**

**Objectives:**

**The objectives of the Hotel Management System are multi-faceted:**

* **Efficiency: Automate routine tasks to save time and resources, enabling staff to focus on guest services and experience.**
* **Accuracy: Eliminate manual errors in booking, billing, and guest details, ensuring precise records and transactions.**
* **Guest Satisfaction: Provide personalized and efficient services to enhance guest satisfaction, encouraging repeat visits and positive reviews.**
* **Operational Optimization: Streamline staff management, room allocation, and inventory tracking for operational excellence.**
* **Financial Management: Ensure accurate billing, payment processing, and financial reporting, aiding in financial decision-making.**
* **Feedback Utilization: Gather and analyze guest feedback to identify areas for improvement, thereby enhancing overall service quality.**
* **Adaptability: Create a flexible system that can be customized to fit the unique needs of different hotels and resorts.**

**By achieving these objectives, the Hotel Management System aims to set new standards in hotel management, ensuring a seamless and delightful experience for both guests and hotel staff.**

|  |  |  |
| --- | --- | --- |
| STAFF | | |
| Field | Type | Key |
| StaffID | Int | PRI |
| FirstName | varchar(50) |  |
| LastName | varchar(50) |  |
| Role | varchar(50) |  |
| Salary | decimal(10,2) |  |
| Email | varchar(100) | UNI |
| PhoneNumber | varchar(15) |  |

|  |  |  |
| --- | --- | --- |
| Events | | |
| Field | Type | Key |
| EventID | Int | PRI |
| EventName | varchar(100) |  |
| EventDate | Date |  |
| Description | Text |  |

**EEERDiagram:**

|  |  |  |
| --- | --- | --- |
| Rooms | | |
| Field | Type | Key |
| RoomID | Int | PRI |
| RoomNumber | varchar(10) | Uni |
| RoomType | varchar(50) |  |
| Price | decimal(10,2) |  |
| IsOccupied | Tinyint(1) |  |

|  |
| --- |
| **Hotel** |

|  |  |  |
| --- | --- | --- |
| Guests | | |
| Field | Type | Key |
| GuestID | Int | PRI |
| FirstName | varchar(50) |  |
| LastName | varchar(50) |  |
| Email | varchar(100) | Uni |
| PHONENUMBER | Bigint |  |
| CheckInDate | Date |  |
| CheckOutDate | Date |  |
| RoomID | Int |  |

|  |  |  |
| --- | --- | --- |
| Reservations | | |
| Field | Type | Key |
| ReservationID | Int | PRI |
| GuestID | Int | MUL |
| RoomID | Int | MUL |
| ReservationDate | Date |  |
| CheckInDate | Date |  |
| CheckOutDate | Date |  |

|  |  |  |
| --- | --- | --- |
| Payments | | |
| Field | Type | Key |
| PaymentID | Int | PRI |
| GuestID | Int | Mul |
| PaymentDate | Date |  |
| Amount | decimal(10,2) |  |
| PaymentMethod | varchar(50) |  |

|  |  |  |
| --- | --- | --- |
| Reviews | | |
| Field | Type | Key |
| ReviewID | Int | PRI |
| GuestID | Int | Mul |
| ReviewDate | Date |  |
| Rating | Int |  |
| Comment | Text |  |

|  |  |  |
| --- | --- | --- |
| Roomservices | | |
| Field | Type | Key |
| ServiceID | Int | PRI |
| GuestID | Int |  |
| ServiceName | varchar(100) |  |
| ServiceDate | date |  |
| Amount | decimal(10,2) |  |

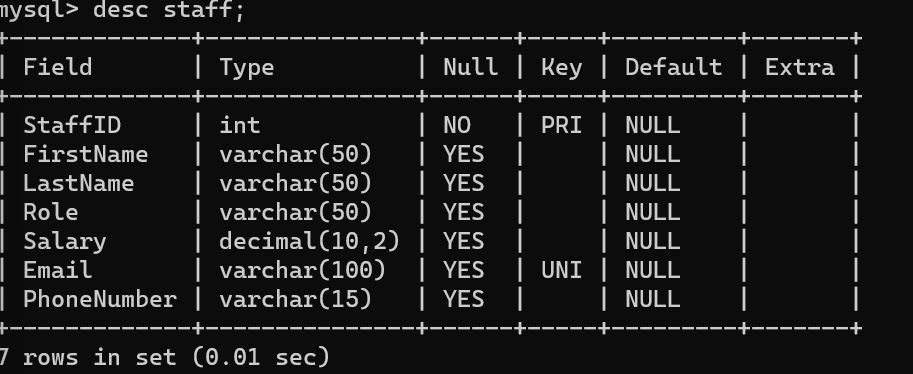
|  |  |  |
| --- | --- | --- |
| Roomcategories | | |
| Field | Type | Key |
| CategoryID | Int | PRI |
| CategoryName | Varchar(50) |  |
| Description | Text |  |

|  |  |  |
| --- | --- | --- |
| Room\_type | | |
| Field | Type | Key |
| RoomTypeID | Int | PRI |
| TypeName | Varchar(50) |  |
| Description | Text |  |

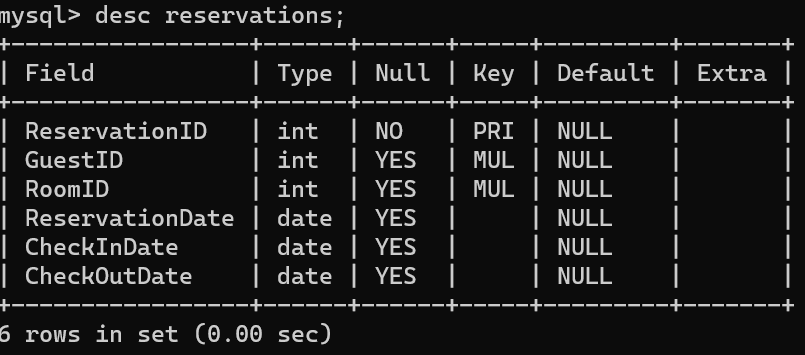
|  |  |  |
| --- | --- | --- |
| Roommaintenance | | |
| Field | Type | Key |
| MaintenanceID | Int | PRI |
| RoomID | INT | Mul |
| MaintenanceDate | Date |  |
| Description | Text |  |

**structure of table:**

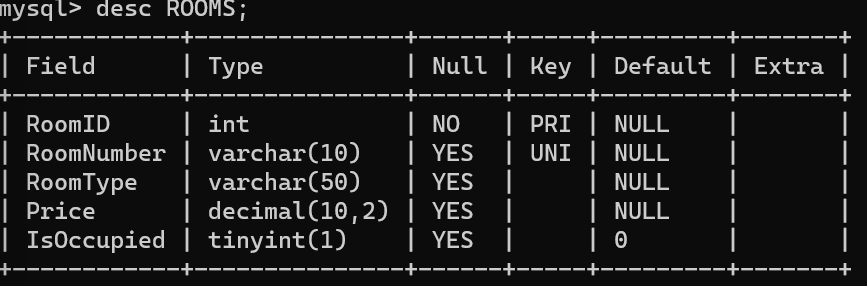
**STAFF:**

****

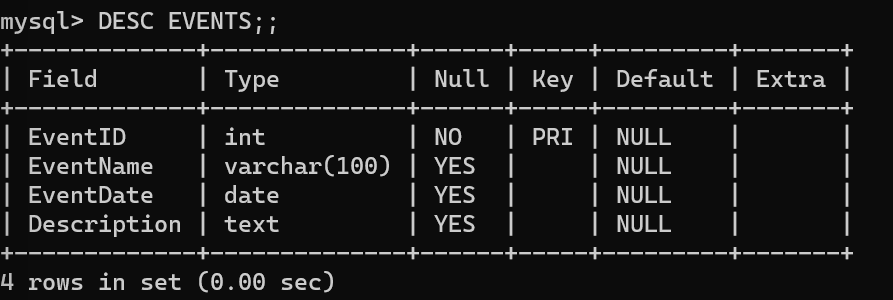
**RESERVATION:**



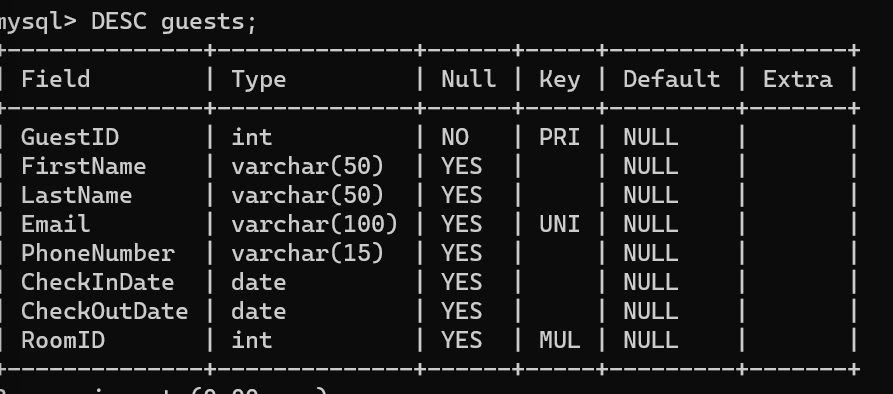
**ROOMS:**

****

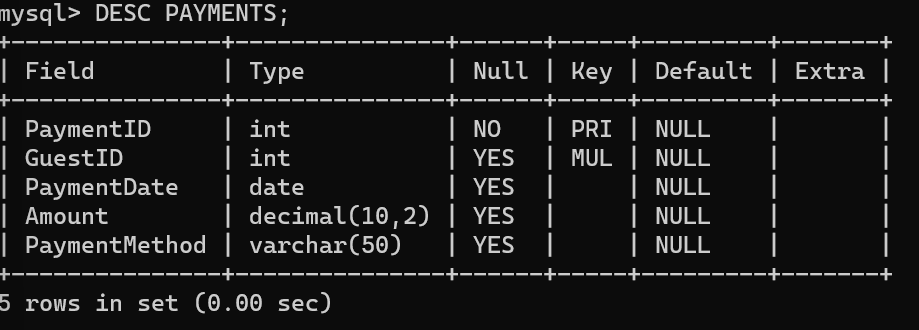
**EVENTS:**

****

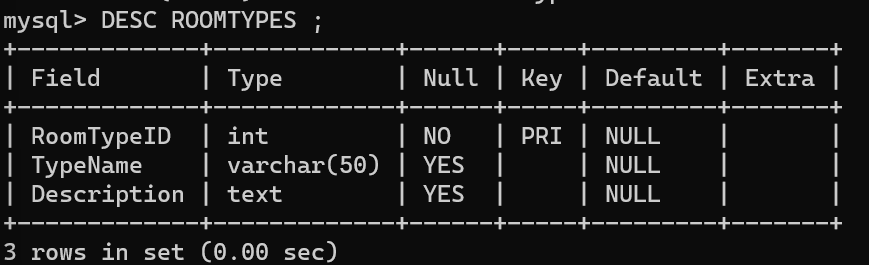
**GUESTS:**

****

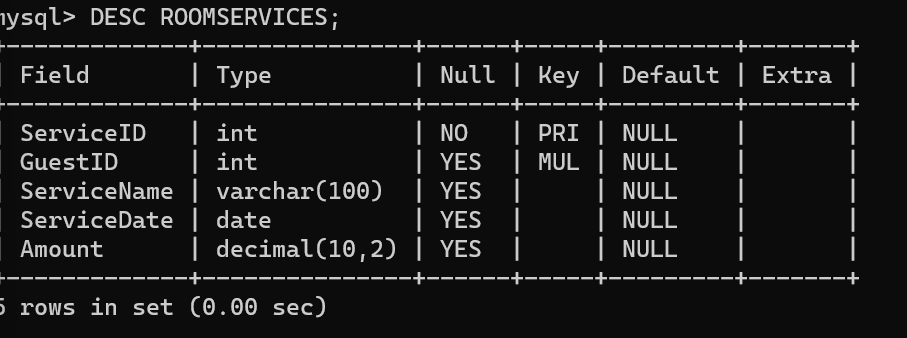
**PAYMENTS:**

****

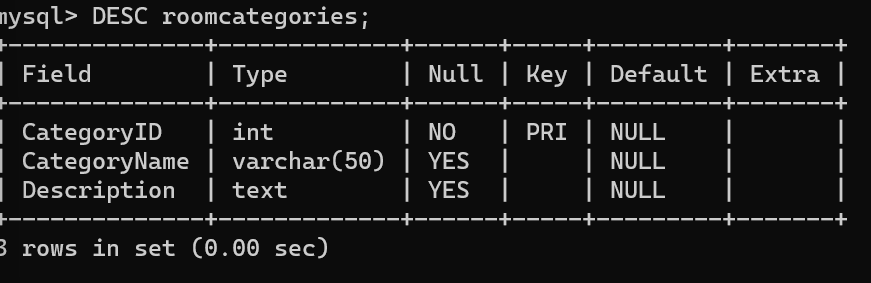
**ROOMTYPE:**

****

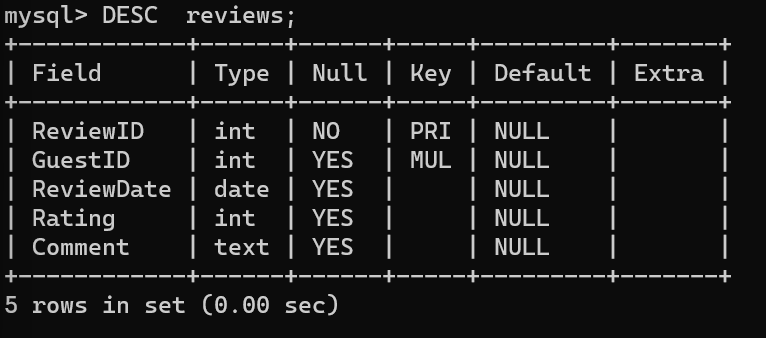
**ROOM SERVICES:**

****

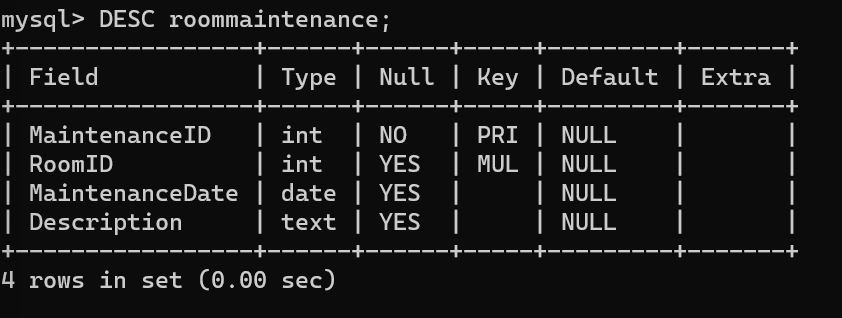
**ROOMCATEGORIES:**

****

**REVIEWS:**

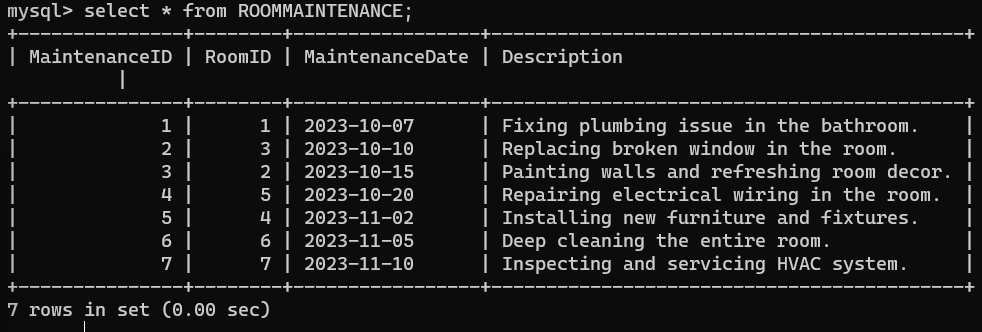
****

**ROOM MAINTENANCE:**

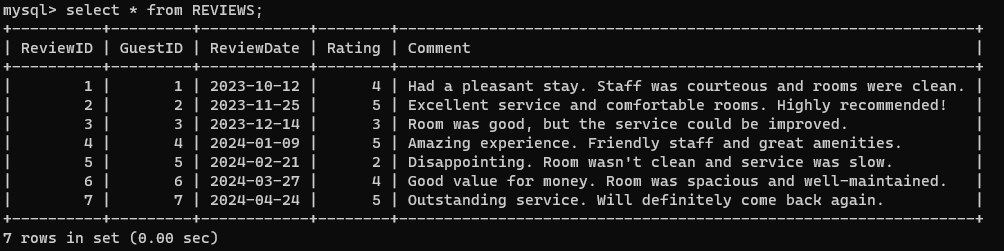
****

**Contents of Tables:**

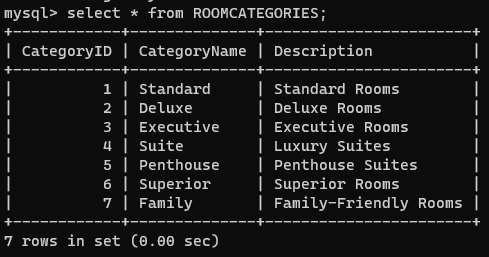
**ROOM MAINTENANCE:**

****

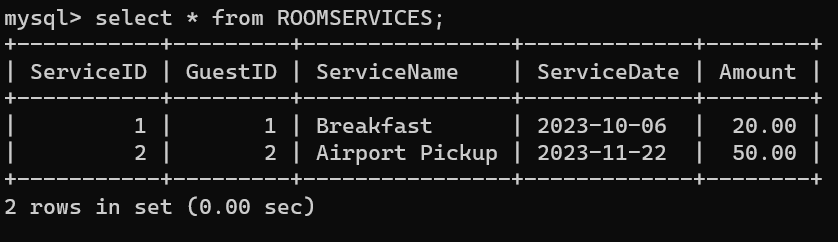
**REVIEWS:**

****

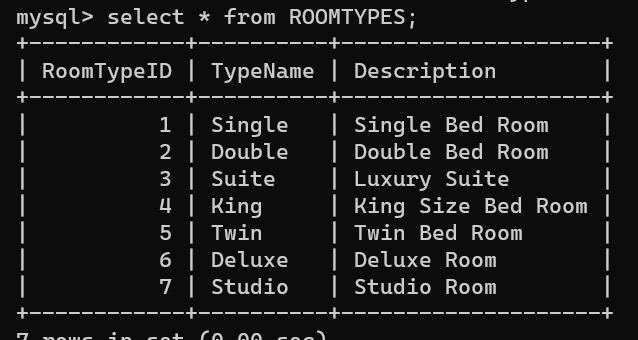
**ROOMCATEGORIES:**:



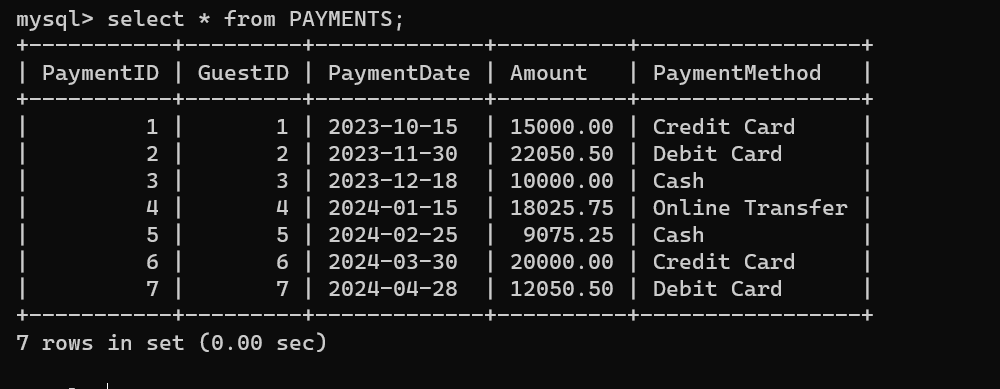
**ROOM SERVICES:**

****

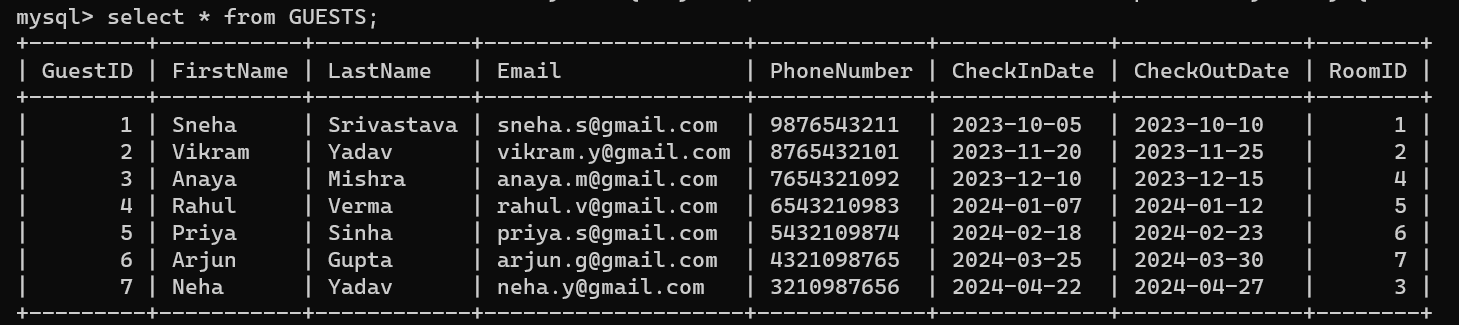
**ROOMTYPE**



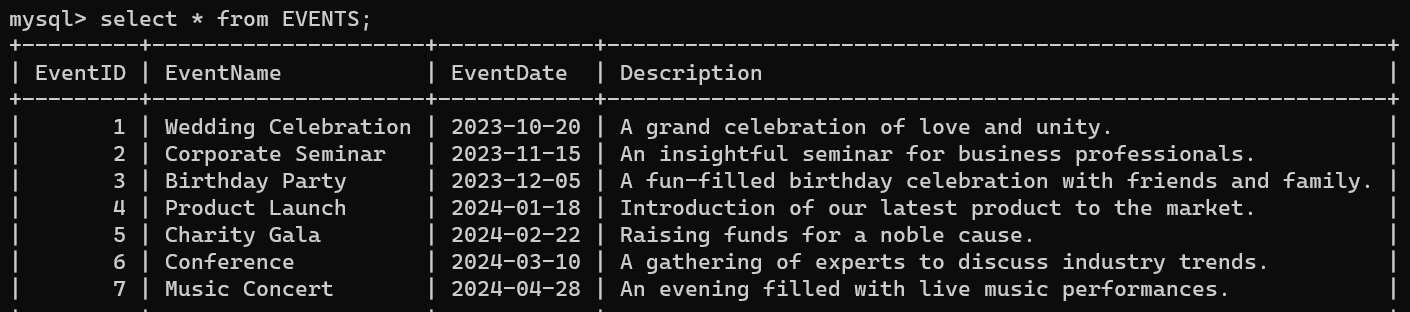
**PAYMENTS**



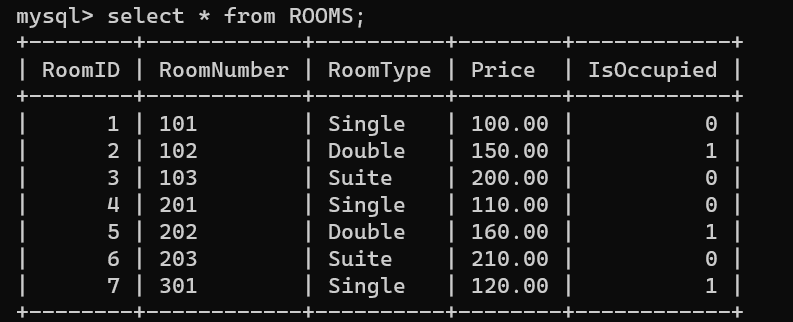
**GUESTS**



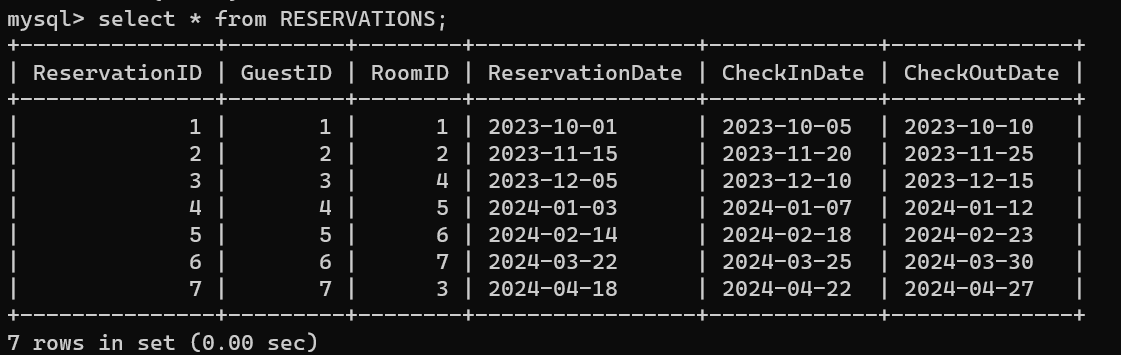
**EVENTS**



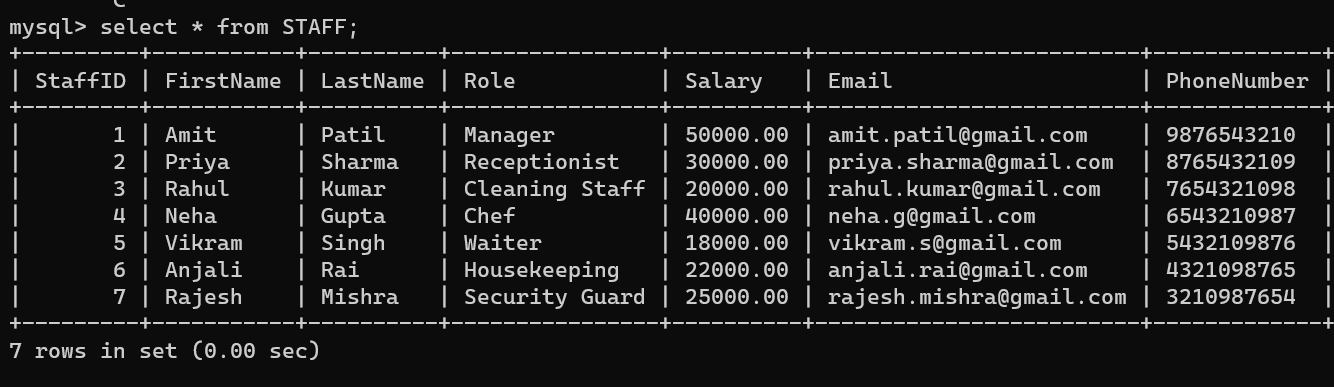
**ROOMS**



**RESERVATION**



**STAFF**

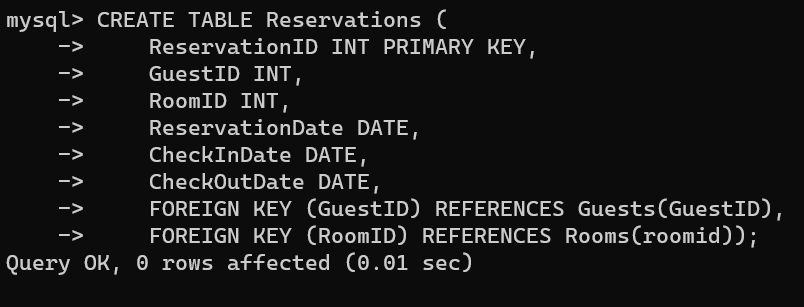


**QUERIES**

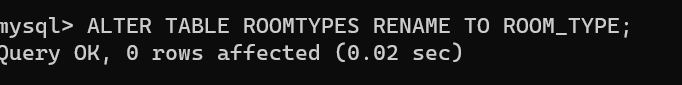
**Here are some questions that cover various aspects of MySQL, including DDL (Data Definition Language), DML (Data Manipulation Language), DQL (Data Query Language), constraints, and clauses. These questions are designed to test your understanding of database concepts and MySQL operations.**

DATA DEFINITION LANGUAGE(DDL):

1.TABLE CREATION:

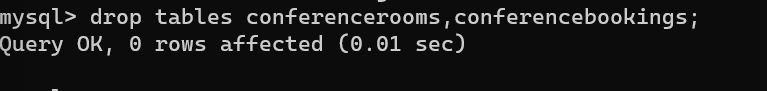


2.ADDING CONSTRAINTS:



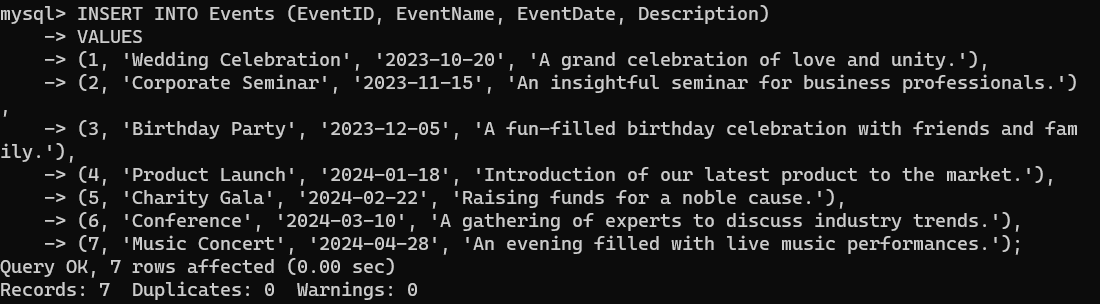
3.MODIFY TABLE:

* **DROP TABLE**

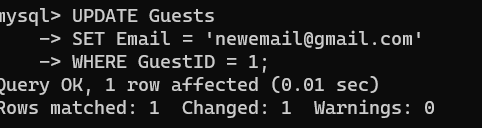
****

DATA MANIPULATION LANGUAGE(DML):

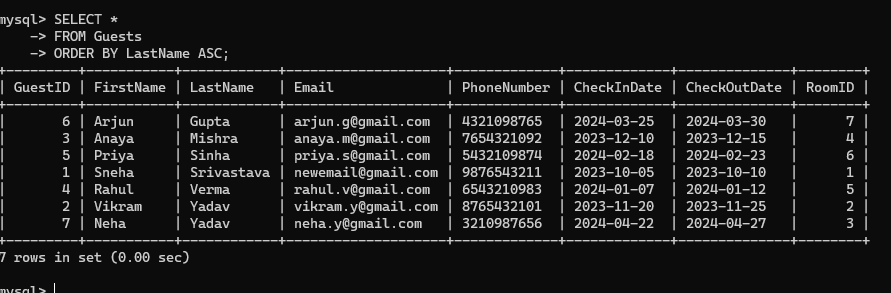
1.INSERT DATA:

****

2.UPDATE DATA

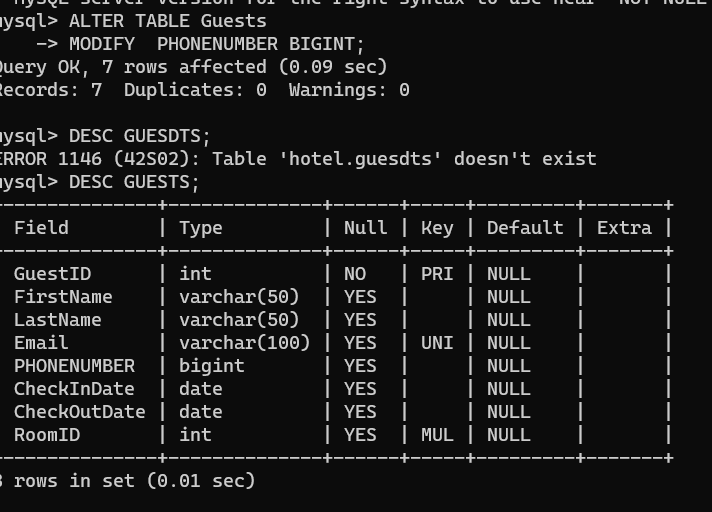
****

**3.ORDERBY DATA**

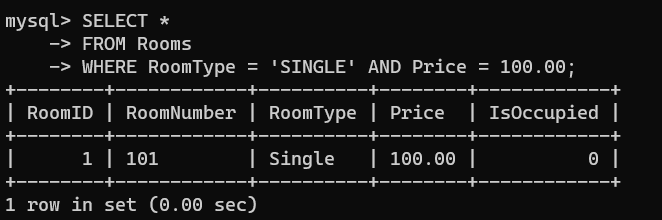
****

Constraints ,Clauses ,AGGREGATE FUNCTION and operators

**Constraints:**

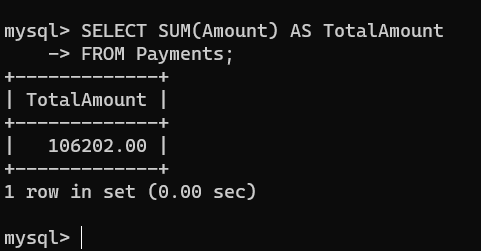


Clauses:

**Using WHERE clause to filter results from the Rooms table.**

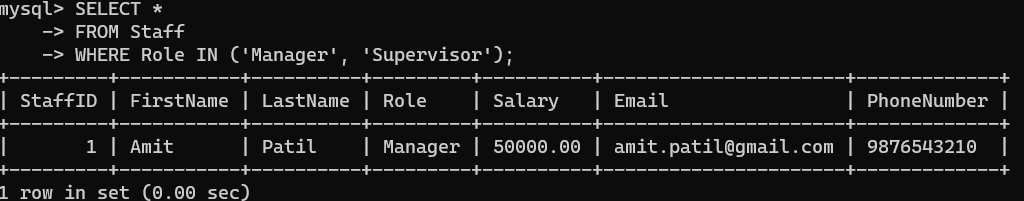
**Aggregate Functions:**

**Using SUM to calculate the total amount spent by all guests**

****

**Operators:**

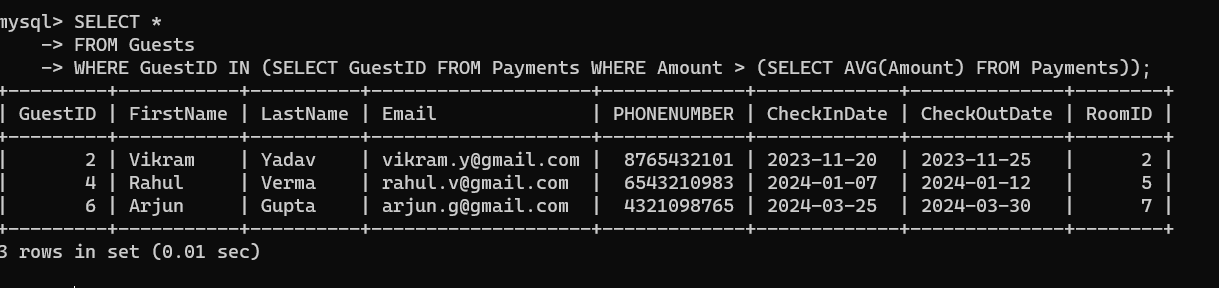
**Using IN operator to filter results from the Staff table.**

****

SUBQUERY AND JOINS

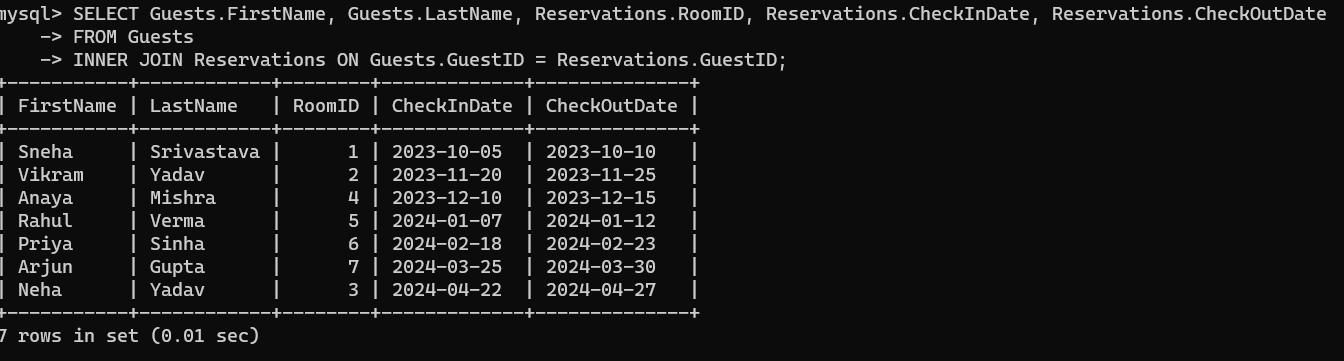
**Subquery:**

**Example 1: Using a subquery to find guests who made payments greater than the average payment.**

****

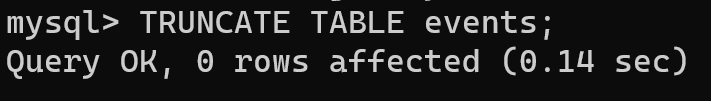
**Joins:**

**Example 2: Using an inner join to combine data from the Guests and Reservations tables.**

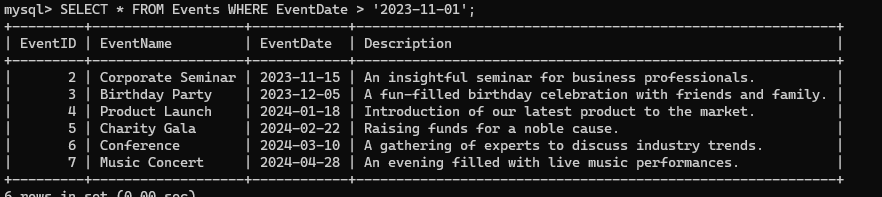
****

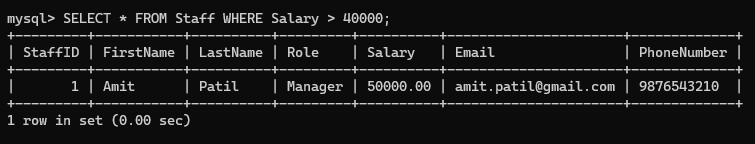
**TCL (Transaction Control Language)**

**• Truncate**



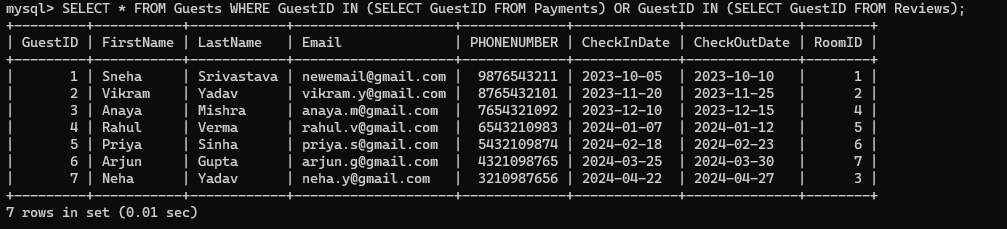
* **COMPARISON**





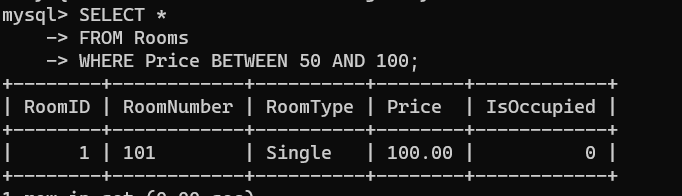
**➢ LOGICAL**

**• OR**

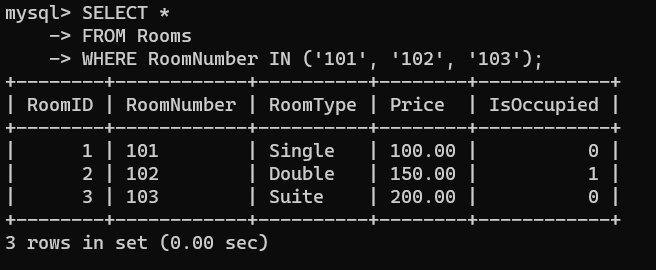


**➢ RANGE, LIKE, LIST**

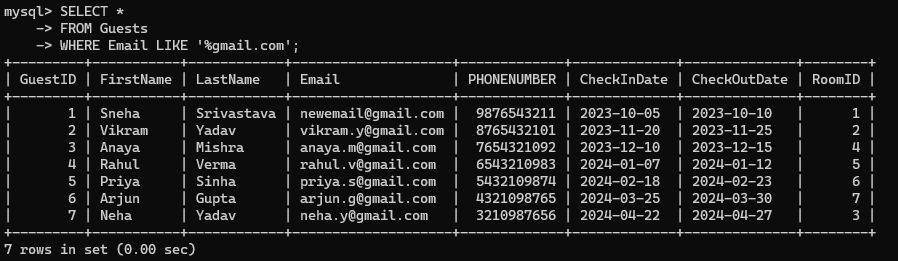
**Using BETWEEN Operator**

****

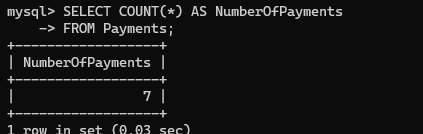
**Using IN Operator:**

****

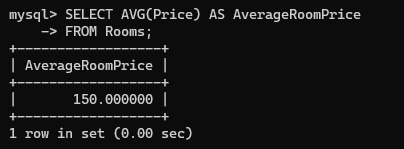
**Using LIKE Operator:**

****

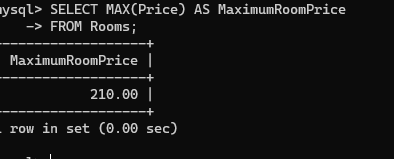
**Using COUNT:**

****

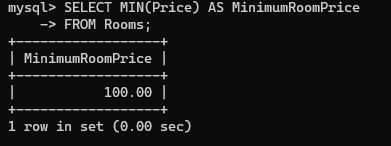
**Using AVG:**

****

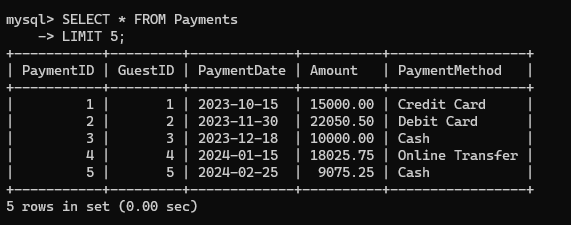
**Using MAX:**

****

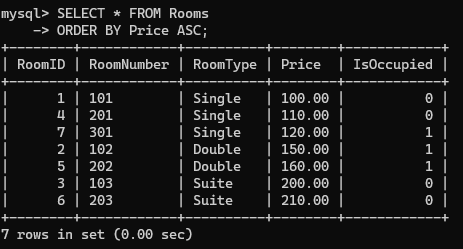
**Using MIN:**

****

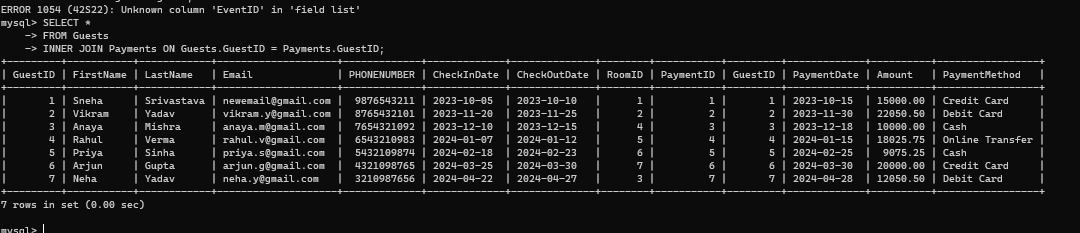
**Using LIMIT:**

****

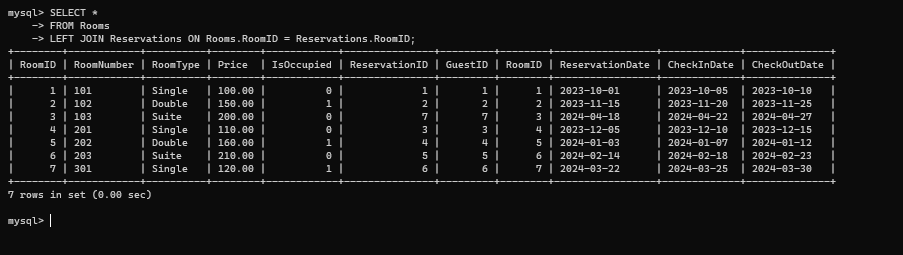
**Using ORDER BY:**

****

**INNER JOIN:**

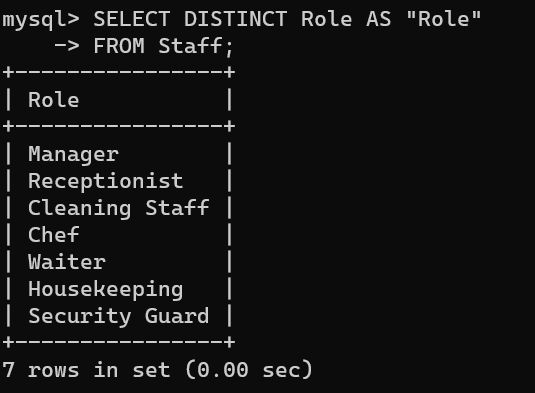
****

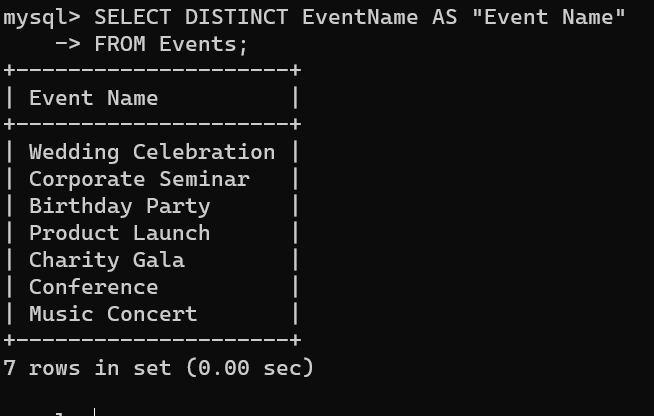
**LEFT JOIN:**

****

**Using DISTINCT and Alias**

****

****

****