

# HOTEL CHAIN MANAGEMENT SYSTEM



**MYSQL QUERIES ON**  
**NORMALIZATION**  
**PROJECT**

**GROUP 16**

**Nimra Ashraf & Noor Fatima**

**110829 & 110830**

**MORNING**

**Submitted to:**

*Mam Sehrish Khan*

**11/05/2025**

# TABLE OF CONTENTS

---

## 1. DESCRIPTION

## 2. ENTITY RELATIONSHIP DIAGRAM

## 3. RELATIONAL SCHEMA

## 4. SQL QUERIES

|                                 |                         |
|---------------------------------|-------------------------|
| ✚ SHOW DATABASES                | ✚ TABLE ALIAS           |
| ✚ CREATING DATABASE &<br>TABLES | ✚ WHERE CLAUSE          |
| ✚ USING DATABASE                | ✚ DISTINCT COMMAND      |
| ✚ DESCRIBE TABLES               | ✚ BETWEEN COMMAND       |
| ✚ SELECT                        | ✚ COUNT                 |
| ✚ INSERT INTO                   | ✚ HAVING                |
| ✚ ALTER TABLES                  | ✚ AND & OR              |
| ✚ TABLE UPDATION                | ✚ IN                    |
| ✚ TABLE CONSTRAINTS             | ✚ LIKE                  |
| ✚ ARITHMETIC OPERATIONS         | ✚ IS NULL & IS NOT NULL |
| ✚ RELATIONAL OPERATIONS         | ✚ DELETE                |
| ✚ AGGREGATE FUNCTIONS           | ✚ DROP,                 |
| ✚ ORDER BY CLAUSE               | ✚ TRUNCATE              |
| ✚ GROUP BY CLAUSE               | ✚ GRANT                 |
|                                 | ✚ PRIVILEGE             |

## 5. NORMALIZATION TABLES

## 6. PRACTICAL IMPLEMENTATION OF QUERIES

# **DESCRIPTION**

SQL stands for structure query language. It is a widely used, open source relational database management system.

A hotel chain management system provides a centralized platform for managing multiple hotel properties within a chain. This database would store information about different hotels, including room details, staff managing, reservations, guest records pricing information etc.

## **Pros:**

- Scalability
- Reliability
- Cost effectiveness
- Flexibility

# ENTITY RELATIONSHIP

## DIAGRAM





# ⇒ RELATIONAL SCHEMA

1. **hotel\_branch** (h\_id, h\_name, h\_address, staff\_id)
2. **staff** (s\_id, s\_name, s\_address, s\_role)
3. **room\_booking** (R\_id, R\_no, check\_in\_date, check\_out\_date, weekend\_pricing, holiday\_pricing, early\_bird\_offers, s\_id)
4. **customer** (c\_id, c\_name, c\_address, r\_id)
5. **loyalty\_points** (points\_id, upgrade\_points, service\_points, cus\_id)
6. **feedback** (f\_id, f\_type, comment, customer\_id)
7. **pre\_order** (order\_id, meal, taxi\_service, cust\_id)
8. **task** (t\_id, type, s\_id)

# SQL QUERIES

## Show DATABASES

---

Show all databases and tables placed in a database

MySQL use the following command:

⇒ **show databases;**

⇒ **show tables;**

## CREATE DATABASE & TABLES

---

Create a new database or table.

⇒ **create database;**

⇒ **create table table\_name (attribute datatype (size),  
...);**

## USING DATABASE

---

Use a database already saved in MySQL.

⇒ **use database database\_name;**

## DESCRIBE TABLES

---

To see the constraints we have assigned to tables,

DESCRIBE keyword is used.

⇒ **DESCRIBE table\_name;**



# SELECT

---

Used to retrieve rows selected from one or more tables.

⇒ **Select \* from table\_name;**

⇒ **Select attribute1\_name from table\_name where attribute2\_name='...' ;**

# INSERT INTO

---

Insert values in tables.

⇒ **INSERT INTO table\_name values ('attribute' datatype (value), ....);**

# ALTER TABLES

---

Used for many purposes such as:

**1.To rename a column of a table**

⇒ **ALTER table table\_name RENAME COLUMN column\_name from existing\_name to new\_name;**

**2.To add a new column in a table**

⇒ **ALTER table table\_name ADD column datatype (size);**

### **3.To make an attribute foreign key**

⇒ **ALTER table table\_name ADD column\_name  
FOREIGN KEY REFERENCES  
referencetable\_name (P.K);**

## **TABLE UPDATION**

---

To reset the values of attributes in a table.

⇒ **Update table\_name set attribute='value' WHERE  
attribute\_PK='target-value';**

## **TABLE CONSTRAINTS**

---

There are two constraints for tables, Primary key and Foreign key.

### **Primary Key:**

The attribute of a table on which all the other attributes of that depend.

### **Foreign Key:**

When Primary key of a table is used in another table, it becomes Foreign key.

⇒ **ALTER table table\_name ADD FOREIGN KEY  
(key\_name) REFERENCES reference\_table (P.K);**

## Arithmetic Operations

---

Arithmetic operations include operators such as +, -, \*, / etc. to perform addition, subtraction, multiplication and division of two attributes.

⇒ **select attribute operator value from table;**

## Logical / Relational Operations

---

Logical operations include operators such as >, <, <=, >=, !=, == etc. to relate any two attributes of a table.

⇒ **select attribute1 from table\_name where attribute2 relational operator value;**

## Aggregation Functions

---

Aggregate functions include avg, max, min etc. operations.

⇒ **select aggregate\_function (attribute) from table;**

## ORDER BY

---

ORDER BY keyword is used to sort the values of



tables in ascending or descending order. By default ascending order is set. Otherwise for sorting the records in descending order DESC order is used. For ascending ASC and descending DESC is used.

Syntaxes of above two orders are:

⇒ **select \* from table ORDER BY attribute DESC;**

⇒ **select \* from table ORDER BY attribute ASC;**

## **GROUP BY**

---

**GROUP BY** clause is very important used to group rows from a table based on the values of one or more column. It is used with aggregate functions like **AVG, MAX, MIN, SUM and COUNT** to perform calculations on grouped data. Also we can perform operations on group within the group.

### **Syntax:**

⇒ **select aggregate\_function (attribute) from table  
GROUP BY attributes;**

## AS

---

The AS keyword in MySQL is used to assign an alias to a table or column, making it easier to reference or improving readability. It allows for temporary renaming with a query, which can simplify complex queries and result sets.

## Where

---

This clause is used to filter records. It is used to extract only those records that fulfill a specified condition.

## DISTINCT

---

The DISTINCT keyword in MySQL is used to remove duplicate records from the results of a SELECT query. It ensures that the query returns only unique values in the specified columns.

### Syntax:

⇒ **select DISTINCT attribute from table;**

## BETWEEN

---



The BETWEEN clause is used to show the values/ contents of the table between a given limit. It filter the result set within a specified range.

### **Syntax:**

⇒ **select column\_name from table where column\_name BETWEEN value1 AND value2;**

## **Count & Count(\*)**

---

Used to count the number of rows in a table. This function counts all rows regardless of whether they contain NULL values.

### **Syntax:**

⇒ **select count (\*) attribute\_name from table;**

⇒ **select count (attribute\_name) from table;**

## **HAVING**

---

The HAVING clause is similar to the WHERE clause but is specifically applied after grouping and aggregation, allowing you to filter on the results of

aggregate functions like COUNT, SUM, AVG, and others.

### **Syntax:**

⇒ **select aggregate\_function (attribute) from table  
GROUP BY attributes HAVING count (attribute) >1;**

## **AND & OR**

---

The AND & OR operators are used to filter records based on more than one conditions:

- The AND operator displays a record if all the conditions separated by AND are true.
- The OR operator displays a record if any of the conditions separated by OR are true.

## **IN**

---

The IN operator allows you to specify multiple values in a where clause. It is a shorthand for multiple OR conditions.

⇒ **SELECT attribute\_name(s) FROM table\_name  
WHERE attribute\_name IN (value1, value2, ...);**

# LIKE

---

The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

## Syntax:

⇒ **SELECT column1. . .FROM table\_name**  
**WHERE columnN LIKE pattern;**

1. The percent sign (%) represents zero, one, or multiple characters
2. The underscore sign ( ) represents one, single character

# IS NULL & IS NOT NULL

---

These keywords are used for checking that the values of attributes checked are **NULL** or not.

## Syntax:

⇒ **select attribute\_name from table where attribute IS**  
**NULL;**



⇒ **select attribute\_name from table where attribute IS NOT NULL;**

## **JOINS**

---

Joins allows to retrieve related data from multiple tables in a single query, avoiding the need for different separate queries. There are multiple types of joins such as inner, right, left etc.

### **Syntax:**

⇒ **Select column\_list from table1 JOIN table2 ON table1.  
column = table1;**

### **INNER JOIN:**

The joins in which both the tables have matching values in them are called inner join.

### **Left Join:**

This join return all rows from the left table and matching rows in right table.

Cross join:

A cross join is type of join that return cartesian product of rows from the tables in the join .

### **Equi join:**

It is join operation in sql that combines two table based on a matching column between them.

### **Right Join:**

This join return all rows from right table and matching rows from left table.

## **VIEWS**

---

A MySQL view is a predefined select query that operates on existing data without duplicating it. A view acts as a virtual table.

### **Syntax:**

```
⇒ create or replace VIEW view_name AS select  
    column1, column2 from table_name where  
    condition;
```

## **DELETE**

---



DELETE statement is used to delete rows in a table.  
It deletes a specific row using where clause.

### **Syntax:**

⇒ **delete from table where column\_name=**  
**'value';**

## **DROP**

---

DROP statement is used to delete the whole table along with table structure, attribute and indexes.

### **Syntax:**

⇒ **drop table table\_name;**

## **TRUNCATE**

---

The truncate statement is used to delete all data in the table not the whole table.

### **Syntax:**

⇒ **truncate table\_name;**

## **Sub Query**

---

Subqueries are also known as inner queries or nested queries. It is embedded inside another query and acts as input or output for that query.

### **Syntax:**

⇒ **Select column1, column2... from table where column operator (select column from another\_table where condition); column1, column2, ...:** The columns you want to retrieve.

## **GRANT**

---

Grant is a statement used to assign privileges to user accounts, allowing them to perform specific actions on database projects.

## **PRIVILEGES**

---

Privileges are the rights or permissions assigned to users that determine what actions they can perform on the database.

# NORMALIZATION TABLES



# CREATE DATABASE HOTEL\_CHAIN

```
MySQL 8.0 Command Line Client
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database hotel_chain;
Query OK, 1 row affected (0.02 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| college |
| hotel_chain |
| information_schema |
| mysql |
| mysqltest |
| performance_schema |
| sys |
+-----+
7 rows in set (0.04 sec)
```

## CREATE Hotel\_Branch Table

```
mysql> use hotel_chain;
Database changed
mysql> create table hotel_branch(h_id INT(5) PRIMARY KEY, h_name varchar(20), h_address varchar(20));
Query OK, 0 rows affected, 1 warning (0.05 sec)

mysql> DESCRIBE hotel_branch;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| h_id | int | NO | PRI | NULL | |
| h_name | varchar(20) | YES | | NULL | |
| h_address | varchar(20) | YES | | NULL | |
+-----+
3 rows in set (0.02 sec)

mysql> select * from hotel_branch;
Empty set (0.01 sec)
```

Activate Windows  
Go to Settings to activate Windows.



# INSERTING VALUES IN HOTEL\_BRANCH

---

```
MySQL 8.0 Command Line Client - Unicode
h_address | varchar(30) | YES | NULL |
3 rows in set (0.00 sec)

mysql> select * from hotel_branch;
+----+-----+-----+
| h_id | h_name | h_address |
+----+-----+-----+
| 1001 | unique | lahore |
| 1002 | five_star | Multan |
| 1003 | Super | Sheikhupura |
+----+-----+-----+
3 rows in set (0.00 sec)
```

## 3 NF OF HOTEL\_BRANCH

---

```
mysql> select h_name , h_address from Hotel_branch;
+-----+-----+
| h_name | h_address |
+-----+-----+
| unique | lahore |
| five_star | Multan |
| Super | Sheikhupura |
+-----+-----+
3 rows in set (0.00 sec)
```

## CREATE STAFF TABLE

---

```
mysql> create table staff(s_id INT(5) PRIMARY KEY, s_name varchar(20),s_address varchar(20),s_role varchar(19),R_id INT(5));
Query OK, 0 rows affected, 2 warnings (0.04 sec)

mysql> DESCRIBE staff;
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| s_id | int | NO | PRI | NULL | |
| s_name | varchar(20) | YES | | NULL | |
| s_address | varchar(20) | YES | | NULL | |
| s_role | varchar(19) | YES | | NULL | |
| R_id | int | YES | | NULL | |
+----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```



## INSERTING VALUES IN STAFF

```
mysql> INSERT INTO staff values('1','haran','multan','manager','10'),('2','ahmad','sheikhpura','sweeper','20'),('3','sohail','lahore','receptionist','30');
Query OK, 3 rows affected (0.02 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select * from staff;
```

| s_id | s_name | s_address  | s_role       | R_id |
|------|--------|------------|--------------|------|
| 1    | haran  | multan     | manager      | 10   |
| 2    | ahmad  | sheikhpura | sweeper      | 20   |
| 3    | sohail | lahore     | receptionist | 30   |

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

## 3RD NORMAL FORM OF STAFF

```
mysql> select s_name, s_address, s
-> -\c
mysql> select s_name,s_address,s_role from staff;
```

| s_name | s_address  | s_role       |
|--------|------------|--------------|
| haran  | multan     | manager      |
| ahmad  | sheikhpura | sweeper      |
| sohail | lahore     | receptionist |

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

## ADD s\_id COLUMN IN HOTEL BRANCH TABLE

```
mysql> ALTER table hotel_branch ADD s_id INT(5);
Query OK, 0 rows affected, 1 warning (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 1
```

```
mysql> DESCRIBE hotel_branch;
```

| Field     | Type        | Null | Key | Default | Extra |
|-----------|-------------|------|-----|---------|-------|
| H_id      | int         | NO   | PRI | NULL    |       |
| h_name    | varchar(25) | YES  |     | NULL    |       |
| h_address | varchar(30) | YES  |     | NULL    |       |
| s_id      | int         | YES  |     | NULL    |       |

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

## CHANGE COLUMN NAME (FROM s\_id TO staff\_id)

---

```
mysql> ALTER table Hotel_branch RENAME COLUMN s_id to staff_id;
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Hotel_branch;
```

| Field     | Type        | Null | Key | Default | Extra |
|-----------|-------------|------|-----|---------|-------|
| H_id      | int         | NO   | PRI | NULL    |       |
| h_name    | varchar(25) | YES  |     | NULL    |       |
| h_address | varchar(30) | YES  |     | NULL    |       |
| staff_id  | int         | YES  |     | NULL    |       |

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

## MAKE staff\_id FOREIGN KEY

---

```
mysql> ALTER table Hotel_branch ADD FOREIGN KEY (staff_id) REFERENCES staff (s_id);
Query OK, 3 rows affected (0.14 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Hotel_branch;
```

| Field     | Type        | Null | Key | Default | Extra |
|-----------|-------------|------|-----|---------|-------|
| H_id      | int         | NO   | PRI | NULL    |       |
| h_name    | varchar(25) | YES  |     | NULL    |       |
| h_address | varchar(30) | YES  |     | NULL    |       |
| staff_id  | int         | YES  | FOR | NULL    |       |

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

## INSERTING VALUES IN FOREIGN KEY

---

```
mysql> Update Hotel_branch set staff_id='1' where H_id='1001';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> Update Hotel_branch set staff_id='2' where H_id='1002';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> Update Hotel_branch set staff_id='3' where H_id='1003';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

## SELECT FROM HOTEL\_BRANCH

---

```
mysql> select * from Hotel_branch;
```

| H_id | h_name    | h_address  | staff_id |
|------|-----------|------------|----------|
| 1001 | unique    | Lahore     | 1        |
| 1002 | five_star | Multan     | 2        |
| 1003 | Super     | Sheikhpura | 3        |

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

## CREATE ROOM\_BOOKING TABLE

---

```
mysql> create table Room_booking(room_id INT(5) PRIMARY KEY, r_no INT(6), check_in_date INT(20), check_out_date INT(20), weekend_pricing INT(20), holiday_pricing INT(19), early_bird_offers INT(25));
Query OK, 0 rows affected, 7 warnings (0.04 sec)
```

```
mysql> DESCRIBE Room_booking;
```

| Field             | Type | Null | Key | Default | Extra |
|-------------------|------|------|-----|---------|-------|
| room_id           | int  | NO   | PRI | NULL    |       |
| r_no              | int  | YES  |     | NULL    |       |
| check_in_date     | int  | YES  |     | NULL    |       |
| check_out_date    | int  | YES  |     | NULL    |       |
| weekend_pricing   | int  | YES  |     | NULL    |       |
| holiday_pricing   | int  | YES  |     | NULL    |       |
| early_bird_offers | int  | YES  |     | NULL    |       |

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

## MODIFY CHECK\_IN & CHECK\_OUT DATES

---

```
mysql> ALTER TABLE Room_booking
  -> MODIFY COLUMN check_in_date DATE;
Query OK, 0 rows affected (0.13 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER table Room_booking MODIFY COLUMN check_in_date DATE;
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER table Room_booking MODIFY COLUMN check_out_date DATE;
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0
```



## DESCRIBE ROOM\_BOOKING

```
mysql> DESCRIBE Room_booking;
```

| Field             | Type | Null | Key | Default | Extra |
|-------------------|------|------|-----|---------|-------|
| room_id           | int  | NO   | PRI | NULL    |       |
| r_no              | int  | YES  |     | NULL    |       |
| check_in_date     | date | YES  |     | NULL    |       |
| check_out_date    | date | YES  |     | NULL    |       |
| weekend_pricing   | int  | YES  |     | NULL    |       |
| hoilday_pricing   | int  | YES  |     | NULL    |       |
| early_bird_offers | int  | YES  |     | NULL    |       |

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

## MODIFY EARLY\_BIRD\_OFFERS & INSERTING VALUES

```
mysql> ALTER table Room_booking MODIFY COLUMN early_bird_offers varchar(29);
```

```
Query OK, 0 rows affected (0.08 sec)
```

```
mysql> INSERT INTO Room_booking values('10','202','2025-01-05','2025-01-25','8000','9000','40%');
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO Room_booking values('20','504','2025-02-4','2025-02-8','7000','8000','30%');
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO Room_booking values('30','301','2025-07-4','2025-07-8','8000','12000','30%');
```

```
Query OK, 1 row affected (0.01 sec)
```

## SELECT FROM ROOM\_BOOKING

```
mysql> select * from Room_booking;
```

| room_id | r_no | check_in_date | check_out_date | weekend_pricing | hoilday_pricing | early_bird_offers |
|---------|------|---------------|----------------|-----------------|-----------------|-------------------|
| 10      | 202  | 2025-01-05    | 2025-01-25     | 8000            | 9000            | 40%               |
| 20      | 504  | 2025-02-04    | 2025-02-08     | 7000            | 8000            | 30%               |
| 30      | 301  | 2025-07-04    | 2025-07-08     | 8000            | 12000           | 30%               |

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

## ADD STAFF\_ID FOREIGN KEY IN ROOM\_BOOKING

```
mysql> ALTER table room_booking ADD FOREIGN KEY (STAFF_ID) REFERENCES staff(s_id);
Query OK, 3 rows affected (0.12 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE room_booking;
```

| Field             | Type        | Null | Key | Default | Extra |
|-------------------|-------------|------|-----|---------|-------|
| room_id           | int         | NO   | PRI | NULL    |       |
| r_no              | int         | YES  |     | NULL    |       |
| check_in_date     | date        | YES  |     | NULL    |       |
| check_out_date    | date        | YES  |     | NULL    |       |
| weekend_pricing   | int         | YES  |     | NULL    |       |
| holiday_pricing   | int         | YES  |     | NULL    |       |
| early_bird_offers | varchar(29) | YES  |     | NULL    |       |
| STAFF_ID          | int         | YES  | MUL | NULL    |       |

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

## INSERT VALUES IN FOREIGN KEY

```
mysql> Update room_booking set STAFF_ID='1' where room_id='10';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> Update room_booking set STAFF_ID='2' where room_id='20';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> Update room_booking set STAFF_ID='3' where room_id='30';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

## SELECT FROM ROOM\_BOOKING

```
mysql> select * from room_booking;
```

| room_id | r_no | check_in_date | check_out_date | weekend_pricing | holiday_pricing | early_bird_offers | STAFF_ID |
|---------|------|---------------|----------------|-----------------|-----------------|-------------------|----------|
| 10      | 202  | 2025-01-05    | 2025-01-25     | 8000            | 9000            | 40%               | 1        |
| 20      | 504  | 2025-02-04    | 2025-02-08     | 7000            | 8000            | 30%               | 2        |
| 30      | 301  | 2025-07-04    | 2025-07-08     | 8000            | 12000           | 30%               | 3        |

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



## 3 NF OF ROOM\_BOOKING

```
mysql> select r_no,weekend_pricing,holiday_pricing,early_bird_offers from Room_booking;
ERROR 1054 (42S22): Unknown column 'holiday_pricing' in 'field list'
mysql> select r_no,weekend_pricing,hoilday_pricing,early_bird_offers from Room_booking;
```

| r_no | weekend_pricing | hoilday_pricing | early_bird_offers |
|------|-----------------|-----------------|-------------------|
| 202  | 8000            | 9000            | 40%               |
| 504  | 7000            | 8000            | 30%               |
| 301  | 8000            | 12000           | 30%               |

3 rows in set (0.00 sec)

## MAKE ROOM\_ID FOREIGN KEY IN STAFF TABLE

```
mysql> ALTER table staff ADD FOREIGN KEY (R_id) REFERENCES Room_booking (room_id);
Query OK, 3 rows affected (0.14 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE staff;
```

| Field     | Type        | Null | Key | Default | Extra |
|-----------|-------------|------|-----|---------|-------|
| s_id      | int         | NO   | PRI | NULL    |       |
| s_name    | varchar(20) | YES  |     | NULL    |       |
| s_address | varchar(20) | YES  |     | NULL    |       |
| s_role    | varchar(19) | YES  |     | NULL    |       |
| R_id      | int         | YES  | MUL | NULL    |       |

6 rows in set (0.00 sec)

```
mysql>
```

## CREATE CUSTOMER TABLE

```
mysql> create table customer(c_id INT(9) primary key,c_name varchar(15), r_id INT(5));
Query OK, 0 rows affected, 2 warnings (0.03 sec)
```

```
mysql> DESCRIBE customer;
```

| Field  | Type        | Null | Key | Default | Extra |
|--------|-------------|------|-----|---------|-------|
| c_id   | int         | NO   | PRI | NULL    |       |
| c_name | varchar(15) | YES  |     | NULL    |       |
| r_id   | int         | YES  |     | NULL    |       |

3 rows in set (0.00 sec)

## ADD MISING C\_ADDRESS COLUMN IN CUSTOMER TABLE

```
mysql> ALTER table customer ADD c_address varchar(30);
Query OK, 0 rows affected (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE customer;
```

| Field     | Type        | Null | Key | Default | Extra |
|-----------|-------------|------|-----|---------|-------|
| c_id      | int         | NO   | PRI | NULL    |       |
| c_name    | varchar(15) | YES  |     | NULL    |       |
| r_id      | int         | YES  |     | NULL    |       |
| c_address | varchar(30) | YES  |     | NULL    |       |

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

## INSERTING VALUES

```
mysql> INSERT INTO customer values('100','NOOR', '10', 'Farooqabad');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO customer values('200','NIMRA', '20', 'sheikhpura'),('300','shanzey', '30', 'sheikhpura');
Query OK, 2 rows affected (0.01 sec)
Records: 2 Duplicates: 0 Warnings: 0
```

```
mysql> select * from student;
```

```
ERROR 1146 (42S02): table 'hotel_chain.student' doesn't exist
```

```
mysql> select * from customer;
```

| c_id | c_name  | r_id | c_address  |
|------|---------|------|------------|
| 100  | NOOR    | 10   | Farooqabad |
| 200  | NIMRA   | 20   | sheikhpura |
| 300  | shanzey | 30   | sheikhpura |

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

## MAKE r\_id FOREIGN KEY IN CUSTOMER TABLE

```
mysql> ALTER table customer ADD FOREIGN KEY (r_id) REFERENCES Room_booking (room_id);
Query OK, 3 rows affected (0.22 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

```
mysql> describe customer;
```

| Field     | Type        | Null | Key | Default | Extra |
|-----------|-------------|------|-----|---------|-------|
| c_id      | int         | NO   | PRI | NULL    |       |
| c_name    | varchar(15) | YES  |     | NULL    |       |
| r_id      | int         | YES  | MUL | NULL    |       |
| c_address | varchar(30) | YES  |     | NULL    |       |

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

## CREATE LOYALTY POINTS TABLE

```
mysql> create table loyalty_points(point_id INT(10) PRIMARY KEY, service_points INT(8), upgrade_points INT(6), cus_id INT(9));
Query OK, 0 rows affected, 4 warnings (0.04 sec)
```

```
mysql> DESCRIBE loyalty_points;
```

| Field          | Type | Null | Key | Default | Extra |
|----------------|------|------|-----|---------|-------|
| point_id       | int  | NO   | PRI | NULL    |       |
| service_points | int  | YES  |     | NULL    |       |
| upgrade_points | int  | YES  |     | NULL    |       |
| cus_id         | int  | YES  |     | NULL    |       |

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

## MAKE cus\_id FOREIGN KEY

```
mysql> ALTER table loyalty_points ADD FOREIGN KEY (cus_id) REFERENCES customer (c_id);
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> describe loyalty_points;
```

| Field          | Type | Null | Key | Default | Extra |
|----------------|------|------|-----|---------|-------|
| point_id       | int  | NO   | PRI | NULL    |       |
| service_points | int  | YES  |     | NULL    |       |
| upgrade_points | int  | YES  |     | NULL    |       |
| cus_id         | int  | YES  | MUL | NULL    |       |

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

## INSERTING VALUES & DESCRIBE

```
mysql> INSERT INTO loyalty_points values('104','300','50','100');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO loyalty_points values('208','500','100','200'),('310','400','60','300');
Query OK, 2 rows affected (0.01 sec)
Records: 2 Duplicates: 0 Warnings: 0
```

```
mysql> select * from loyalty_points;
```

| point_id | service_points | upgrade_points | cus_id |
|----------|----------------|----------------|--------|
| 104      | 300            | 50             | 100    |
| 208      | 500            | 100            | 200    |
| 310      | 400            | 60             | 300    |

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



## CREATE FEEDBACK TABLE

```
mysql> create table feedback(fb_id INT(15) PRIMARY KEY, fb_type varchar(20), comment varchar(30), customer_id INT(10));
Query OK, 0 rows affected, 2 warnings (0.04 sec)
```

```
mysql> DESCRIBE feedback;
```

| Field       | Type        | Null | Key | Default | Extra |
|-------------|-------------|------|-----|---------|-------|
| fb_id       | int         | NO   | PRI | NULL    |       |
| fb_type     | varchar(20) | YES  |     | NULL    |       |
| comment     | varchar(30) | YES  |     | NULL    |       |
| customer_id | int         | YES  |     | NULL    |       |

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

## MAKE customer\_id FOREIGN KEY IN FEEDBACK TABLE

```
mysql> INSERT INTO feedback values('203','compliment','great food','100');
Query OK, 1 row affected (0.02 sec)
```

```
mysql> INSERT INTO feedback values('202','complaint','late services','200');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO feedback values('300','compliment','good services','300');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from feedback;
```

| fb_id | fb_type    | comment       | customer_id |
|-------|------------|---------------|-------------|
| 202   | complaint  | late services | 200         |
| 203   | compliment | great food    | 100         |
| 300   | compliment | good services | 300         |

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

## INSERTING VALUES

```
mysql> ALTER table feedback ADD FOREIGN KEY (customer_id) REFERENCES customer (c_id);
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> describe feedback;
```

| Field       | Type        | Null | Key | Default | Extra |
|-------------|-------------|------|-----|---------|-------|
| fb_id       | int         | NO   | PRI | NULL    |       |
| fb_type     | varchar(20) | YES  |     | NULL    |       |
| comment     | varchar(30) | YES  |     | NULL    |       |
| customer_id | int         | YES  | MUL | NULL    |       |

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



## CREATE PRE\_ORDER TABLE

```
mysql> create table PreOrder(order_id INT(30) PRIMARY KEY,meal varchar(40),taxi_services varchar(10),cust_id INT(10));
Query OK, 0 rows affected, 2 warnings (0.04 sec)
```

```
mysql> DESCRIBE PreOrder;
```

| Field         | Type        | Null | Key | Default | Extra |
|---------------|-------------|------|-----|---------|-------|
| order_id      | int         | NO   | PRI | NULL    |       |
| meal          | varchar(40) | YES  |     | NULL    |       |
| taxi_services | varchar(10) | YES  |     | NULL    |       |
| cust_id       | int         | YES  |     | NULL    |       |

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

## MADE cus\_id FOREIGN KEY

```
mysql> ALTER table PreOrder ADD FOREIGN KEY (cust_id) REFERENCES customer (c_id);
```

```
Query OK, 0 rows affected (0.15 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> describe PreOrder;
```

| Field         | Type        | Null | Key | Default | Extra |
|---------------|-------------|------|-----|---------|-------|
| order_id      | int         | NO   | PRI | NULL    |       |
| meal          | varchar(40) | YES  |     | NULL    |       |
| taxi_services | varchar(10) | YES  |     | NULL    |       |
| cust_id       | int         | YES  | MUL | NULL    |       |

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

## INSERTING VALUES IN ORDER

```
mysql> INSERT INTO PreOrder values('110','breakfast','yes','100');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO PreOrder values('220','Dinner','no','200');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> INSERT INTO PreOrder values('330','lunch','yes','300');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from PreOrder;
```

| order_id | meal      | taxi_services | cust_id |
|----------|-----------|---------------|---------|
| 110      | breakfast | yes           | 100     |
| 220      | Dinner    | no            | 200     |
| 330      | lunch     | yes           | 300     |

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

# CREATE TASK TABLE

---

```
mysql> create table task(t_id INT(5) PRIMARY KEY, type varchar(20), st_id INT(5));
Query OK, 0 rows affected, 2 warnings (0.05 sec)
```

```
mysql> DESCRIBE task;
ERROR 1146 (42S02): Table 'hotel_chain.task' doesn't exist
mysql> DESCRIBE task;
```

| Field | Type        | Null | Key | Default | Extra |
|-------|-------------|------|-----|---------|-------|
| t_id  | int         | NO   | PRI | NUL     |       |
| type  | varchar(20) | YES  |     | NUL     |       |
| st_id | int         | YES  |     | NUL     |       |

3 rows in set (0.00 sec)

## ADD st\_id FOREIGN KEY AND DESCRIBE IT

---

```
mysql> ALTER table task ADD FOREIGN KEY (st_id) REFERENCES staff(s_id);
Query OK, 0 rows affected (0.10 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE task;
```

| Field | Type        | Null | Key | Default | Extra |
|-------|-------------|------|-----|---------|-------|
| t_id  | int         | NO   | PRI | NUL     |       |
| type  | varchar(20) | YES  |     | NUL     |       |
| st_id | int         | YES  | MUL | NUL     |       |

3 rows in set (0.00 sec)

## INSERTING VALUES IN TASK

---

```
mysql> INSERT INTO task values('100','cleaning','1');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO task values('201','maintenance','2');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO task values('405','guard','3');
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from task;
```

| t_id | type        | st_id |
|------|-------------|-------|
| 100  | cleaning    | 1     |
| 201  | maintenance | 2     |
| 405  | guard       | 3     |

# **PRACTICAL IMPLEMENTATION OF MYSQL QUERIES**

# ARITHMETIC & LOGICAL OPERATIONS ON LOYALTY\_POINTS TABLE

```
MySQL 8.0 Command Line Client
> rows in set (0.00 sec)

mysql> select service_points*2 from loyalty_points;
+-----+
| service_points*2 |
+-----+
| 000 |
| 2000 |
| 000 |
+-----+
3 rows in set (0.00 sec)

mysql> select upgrade_points/20 from loyalty_points;
+-----+
| upgrade_points/20 |
+-----+
| 10 |
| 100 |
| 20 |
+-----+
3 rows in set (0.00 sec)

mysql> select upgrade_points*50 from loyalty_points;
+-----+
| upgrade_points*50 |
+-----+
| 500 |
| 1000 |
| 100 |
+-----+
3 rows in set (0.00 sec)

mysql> select service_points+100 AS annual from loyalty_points where point_id='200';
+-----+
| annual |
+-----+
| 000 |
+-----+
1 row in set (0.00 sec)

mysql> select upgrade_points+30 AS annual from loyalty_points where point_id='310';
+-----+
| annual |
+-----+
| 30 |
+-----+
1 row in set (0.00 sec)
```

# LOGICAL OPERATIONS ADDING ph\_no COLUMN IN CUSTOMER

```
MySQL 8.0 Command Line Client
+-----+
| 110 |
+-----+
1 row in set (0.00 sec)

mysql> select service_points+100 AS annual from loyalty_points where point_id='200';
+-----+
| annual |
+-----+
| 000 |
+-----+
1 row in set (0.00 sec)

mysql> select upgrade_points+30 AS annual from loyalty_points where point_id='310';
+-----+
| annual |
+-----+
| 30 |
+-----+
1 row in set (0.00 sec)

mysql> select upgrade_points*20+50 from loyalty_points;
+-----+
| upgrade_points*20+50 |
+-----+
| 1010 |
| 2050 |
| 1210 |
+-----+
3 rows in set (0.00 sec)

mysql> select upgrade_points*10+50 AS annual from loyalty_points;
+-----+
| annual |
+-----+
| 550 |
| 1050 |
| 650 |
+-----+
3 rows in set (0.00 sec)

mysql>
```



```

MySQL 8.0 Command Line Client
3 rows in set (0.00 sec)

mysql> ALTER table Customer ADD COLUMN ph_no INT(14);
Query OK, 0 rows affected, 1 warning (0.04 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> UPDATE Customer set ph_no='0323373781' where c_id='100';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> UPDATE Customer set ph_no='0334874849' where c_id='200';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> UPDATE Customer set ph_no='031940083' where c_id='300';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from Customer;
+----+-----+----+-----+-----+
| c_id | c_name | r_id | c_address | ph_no |
+----+-----+----+-----+-----+
| 100 | Shreey | 10 | Jandiyalla | 0323373781 |
| 200 | Naor | 20 | Farooqabad | 0334874849 |
| 300 | Nlara | 30 | Sheikhupura | 031940083 |
+----+-----+----+-----+-----+
3 rows in set (0.00 sec)

```

```

MySQL 8.0 Command Line Client
v_points where service_points>=000' at line 1
mysql> select point_id from loyalty_points where service_points>=000;
+-----+
| point_id |
+-----+
| 200 |
| 310 |
+-----+
2 rows in set (0.00 sec)

mysql> select * from room_booking;
+----+-----+-----+-----+-----+-----+-----+-----+
| room_id | r_no | checkin_date | checkout_date | weekend_pricing | holiday_pricing | earlybird_offers | sta_id |
+----+-----+-----+-----+-----+-----+-----+-----+
| 10 | 202 | 2025-01-22 | 2025-01-26 | 8000 | 9500 | 45% | 1 |
| 20 | 504 | 2025-01-12 | 2025-03-19 | 6000 | 8000 | 60% | 2 |
| 30 | 301 | 2025-07-04 | 2025-07-08 | 8000 | 12000 | 30% | 3 |
+----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from room_booking where weekend_pricing>7000 && holiday_pricing >9000;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'from room_booking where weekend_pricing>7000 && holiday_pricing >9000' at line 1
mysql> select * from room_booking where weekend_pricing>7000 && holiday_pricing >9000;
+----+-----+-----+-----+-----+-----+-----+-----+
| room_id | r_no | checkin_date | checkout_date | weekend_pricing | holiday_pricing | earlybird_offers | sta_id |
+----+-----+-----+-----+-----+-----+-----+-----+
| 10 | 202 | 2025-01-22 | 2025-01-26 | 8000 | 9500 | 45% | 1 |
| 30 | 301 | 2025-07-04 | 2025-07-08 | 8000 | 12000 | 30% | 3 |
+----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set, 1 warning (0.00 sec)

mysql> select * from Customer;
+----+-----+----+-----+
| c_id | c_name | r_id | c_address |
+----+-----+----+-----+
| 100 | Shreey | 10 | Jandiyalla |
| 200 | Naor | 20 | Farooqabad |
| 300 | Nlara | 30 | Sheikhupura |
+----+-----+----+-----+

```

Activate Windows  
Go to Settings to activate Windows.

# ORDER BY CLAUSE ON CUSTOMER

```
MySQL 8.0 Command Line Client
mysql> select * from Customer;
+----+-----+-----+-----+-----+
| c_id | c_name | c_id | c_address | ph_no |
+----+-----+-----+-----+-----+
| 100 | Hiera | 30 | Sheikhupura | 335040083 |
| 200 | Boor | 20 | Feroqabad | 334874849 |
| 300 | Shanzay | 10 | Jandiyalla | 323373781 |
+----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from Customer order by c_id DESC;
+----+-----+-----+-----+-----+
| c_id | c_name | c_id | c_address | ph_no |
+----+-----+-----+-----+-----+
| 300 | Shanzay | 10 | Jandiyalla | 323373781 |
| 200 | Boor | 20 | Feroqabad | 334874849 |
| 100 | Hiera | 30 | Sheikhupura | 335040083 |
+----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from Customer order by c_id ASC;
+----+-----+-----+-----+-----+
| c_id | c_name | c_id | c_address | ph_no |
+----+-----+-----+-----+-----+
| 100 | Shanzay | 10 | Jandiyalla | 323373781 |
| 200 | Boor | 20 | Feroqabad | 334874849 |
| 300 | Hiera | 30 | Sheikhupura | 335040083 |
+----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select c_name from Customer where c_id=200;
+-----+
| c_name |
+-----+
| Boor |
+-----+
1 row in set (0.00 sec)
```

# GIVING TWO h\_ids SAME NAME

```
MySQL 8.0 Command Line Client
mysql> select * from hotel_branch;
+----+-----+-----+-----+
| h_id | h_name | h_address | staff_id |
+----+-----+-----+-----+
| 1001 | unique | Lahore | 1 |
| 1002 | Five_star | Multan | 2 |
| 1003 | super | Sheikhupura | 3 |
+----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE hotel_branch set h_id='1003' where h_name='five_star';
ERROR 1062 (23000): Duplicate entry '1003' for key 'hotel_branch.PRIMARY'
mysql> UPDATE hotel_branch set h_name=NULL where h_id='1003';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from hotel_branch;
+----+-----+-----+-----+
| h_id | h_name | h_address | staff_id |
+----+-----+-----+-----+
| 1001 | unique | Lahore | 1 |
| 1002 | Five_star | Multan | 2 |
| 1003 | NULL | Sheikhupura | 3 |
+----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE hotel_branch set h_name='Five_star' where h_id='1003';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from hotel_branch;
+----+-----+-----+-----+
| h_id | h_name | h_address | staff_id |
+----+-----+-----+-----+
| 1001 | unique | Lahore | 1 |
| 1002 | Five_star | Multan | 2 |
| 1003 | Five_star | Sheikhupura | 3 |
+----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

# DISTINCT CLAUSE AND ADDING SALARY COLUMN IN STAFF TABLE

```
MySQL 8.0 Command Line Client
mysql> select DISTINCT h_name from hotel_branch;
+-----+
| h_name |
+-----+
| unique |
| five_star |
+-----+
2 rows in set (0.01 sec)

mysql> select * from staff;
+----+-----+-----+-----+----+
| s_id | s_name | s_address | s_role | r_id |
+----+-----+-----+-----+----+
| 1 | Haroon | Multan | manager | 18 |
| 2 | Ahmad | Shalidpur | sweeper | 28 |
| 3 | Sohail | Lahore | receptionist | 38 |
+----+-----+-----+-----+----+
3 rows in set (0.00 sec)

mysql> ALTER table staff ADD COLUMN salary INT(10);
Query OK, 0 rows affected, 1 warning (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> DESCRIBE staff;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| s_id | int | NO | PRI | NULL | |
| s_name | varchar(20) | YES | | NULL | |
| s_address | varchar(20) | YES | | NULL | |
| s_role | varchar(20) | YES | | NULL | |
| r_id | int | YES | | NULL | |
| salary | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> UPDATE staff set salary=50000 where s_id=1;
Query OK, 1 row affected (0.01 sec)
```

# BETWEEN, HAVING AND IN CLAUSES

```
MySQL 8.0 Command Line Client - llxcode
mysql> select em_name from employee where job='assistant' OR salary BETWEEN 15000 AND 30000;
+-----+
| em_name |
+-----+
| Ali |
| Ahmad |
| Arhan |
+-----+
3 rows in set (0.00 sec)

mysql> select em_name from Employee where job='manager' OR salary BETWEEN 15000 AND 30000;
+-----+
| em_name |
+-----+
| Ali |
| Ahmad |
| Arhan |
| Rehman |
+-----+
4 rows in set (0.00 sec)

mysql> select em_name from Employee where salary IN (15000,30000);
+-----+
| em_name |
+-----+
| Ahmad |
| Arhan |
+-----+
2 rows in set (0.00 sec)

mysql> select em_name from Employee where salary IN (15000,30000,20000);
+-----+
| em_name |
+-----+
| Ahmad |
| Arhan |
+-----+
2 rows in set (0.00 sec)
```

```
Select MySQL 8.0 Command Line Client - Windows
1 row in set (0.00 sec)

mysql> select max(salary) from staff GROUP BY s_role HAVING count(s_id)>1;
+-----+
| max(salary) |
+-----+
| 13000       |
+-----+
1 row in set (0.00 sec)

mysql> select min(salary) from staff GROUP BY s_role HAVING count(s_id)>1;
+-----+
| min(salary) |
+-----+
| 12000       |
+-----+
1 row in set (0.00 sec)

mysql> select s_name from staff where salary<13000 between salary>8000;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '>8000' at line 1
mysql> select s_name from staff where salary<13000 between salary>8000;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '>8000' at line 1
mysql> select s_name from staff where salary>8000 && salary<13000;
+-----+
| s_name |
+-----+
| haran  |
+-----+
1 row in set, 1 warning (0.00 sec)

mysql> select s_name from staff where s_role='sweeper' && salary>8000 && salary<13000;
Empty set, 2 warnings (0.00 sec)

mysql> select s_name from staff where s_role='sweeper' && salary>=8000 && salary<13000;
+-----+
| s_name |
+-----+
| Ahmad  |
+-----+
1 row in set (0.00 sec)
```

```
MySQL 8.0 Command Line Client
mysql> select min(salary) from staff GROUP BY s_role HAVING count(s_id)>1;
+-----+
| min(salary) |
+-----+
| 40000       |
+-----+
1 row in set (0.00 sec)

mysql> select s_name from staff where s_role='sweeper' && salary<40000;
+-----+
| s_name |
+-----+
| Ahmad  |
+-----+
1 row in set, 1 warning (0.00 sec)

mysql> select s_name from staff where s_role='manager' && salary BETWEEN 35000 AND 50000;
+-----+
| s_name |
+-----+
| haran  |
| Sahil  |
+-----+
2 rows in set, 1 warning (0.00 sec)

mysql> select s_name from staff where s_address='Sheikhpura' && salary BETWEEN 25000 AND 35000;
+-----+
| s_name |
+-----+
| Ahmad  |
+-----+
1 row in set, 1 warning (0.00 sec)

mysql> select s_name from staff where s_address='Sheikhpura' OR salary BETWEEN 25000 AND 35000;
+-----+
| s_name |
+-----+
| Ahmad  |
+-----+
1 row in set (0.00 sec)
```

Activate Windows  
Go to Settings to activate Windows.



```

MySQL 8.0 Command Line Client
1 row in set (0.00 sec)

mysql> select * from room_booking;
+-----+-----+-----+-----+-----+-----+-----+-----+
| room_id | r_no | checkin_date | checkout_date | weekend_pricing | holiday_pricing | earlybird_offers | sta_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10      | 202  | 2025-01-22   | 2025-01-26   | 8000           | 9500            | 45%              | 1      |
| 20      | 504  | 2025-03-12   | 2025-03-19   | 6000           | 8000            | 60%              | 2      |
| 30      | 301  | 2025-07-05   | 2025-07-08   | 8000           | 12000           | 100%             | 3      |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select room_id from room_booking where weekend_pricing IN(8000,9000);
+-----+
| room_id |
+-----+
| 10      |
| 20      |
| 30      |
+-----+
1 row in set (0.00 sec)

mysql> select room_no from room_booking where holiday_pricing IN(12000,9000);
ERROR 1054 (42S22): Unknown column 'room_no' in 'field list'
mysql> select r_no from room_booking where holiday_pricing IN(12000,9000);
+-----+
| r_no   |
+-----+
| 504    |
| 301    |
+-----+
2 rows in set (0.00 sec)

mysql> select * from customer;
+-----+-----+-----+-----+-----+
| c_id | c_name | r_id | c_address | ph_no |
+-----+-----+-----+-----+-----+
| 100  | Shazay | 10   | Jandiyalla | 323371791 |
| 200  | Noor  | 20   | Farooqabad | 374874849 |
| 300  | Wira  | 30   | Shelkhpura | 315940853 |
+-----+-----+-----+-----+-----+

```

# ARITHMETIC OPERATIONS ON STAFF

```

MySQL 8.0 Command Line Client
1 row in set (0.00 sec)

mysql> UPDATE staff set salary='50000' where s_id='1';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> UPDATE staff set salary='30000' where s_id='2';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> UPDATE staff set salary='40000' where s_id='3';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from salary;
ERROR 1106 (42S02): table 'hotel_chain.salary' doesn't exist
mysql> select * from staff;
+-----+-----+-----+-----+-----+
| s_id | s_name | s_address | s_role | r_id | salary |
+-----+-----+-----+-----+-----+
| 1     | Harun  | Pultan   | manager | 10   | 50000  |
| 2     | Ahmad | Shelkhpura | sweeper | 20   | 30000  |
| 3     | Sahal | Lahore   | receptionist | 30   | 40000  |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select avg(salary) from staff;
+-----+
| avg(salary) |
+-----+
| 40000.0000  |
+-----+
1 row in set (0.01 sec)

mysql> select max(salary) from staff;
+-----+
| max(salary) |
+-----+
| 50000       |
+-----+

```

# COUNT & COUNT(\*)

```
MySQL 8.0 Command Line Client - Unicode
mysql> select * from staff_view;
+-----+
10 rows in set (0.00 sec)

mysql> select * from feedback;
+----+-----+-----+-----+
fb_id | fb_type | comment | customer_id |
+----+-----+-----+-----+
202  | complaint | late services | 200 |
203  | complaint | great food | 100 |
300  | complaint | good services | 300 |
+----+-----+-----+-----+
3 rows in set (0.02 sec)

mysql> select count(*) from feedback;
+-----+
count(*) |
+-----+
3 |
+-----+
1 row in set (0.00 sec)

mysql> select count(comment) from feedback;
ERROR 1146 (42S02): table 'hotel_chain.feedback' doesn't exist
mysql> select count(comment) from feedback;
+-----+
count(comment) |
+-----+
3 |
+-----+
1 row in set (0.00 sec)

mysql>
```

# VIEW

```
MySQL 8.0 Command Line Client - Unicode
mysql> create or replace view preorder_view AS select meal,taxi_services from preorder;
ERROR 1146 (42S02): table 'hotel_chain.preorder' doesn't exist
mysql> create or replace view preorder_view AS select meal,taxi_services from preorder;
Query OK, 0 rows affected (0.02 sec)

mysql> select * from preorder_view;
+-----+-----+
meal | taxi_services |
+-----+-----+
breakfast | yes |
dinner | no |
lunch | yes |
+-----+-----+
3 rows in set (0.01 sec)

mysql> create or replace view customer_view AS select c_address,c_name from customer;
Query OK, 0 rows affected (0.02 sec)

mysql> select * from customer_view;
+-----+-----+
c_id | c_name | f_id | c_address | phone_no |
+-----+-----+
100 | NOOR | 10 | Farooqabad | 98876257 |
200 | NIMRA | 20 | Sheikhpura | 38887257 |
300 | shanze | 30 | Sheikhpura | 333447257 |
+-----+-----+
3 rows in set (0.02 sec)

mysql> select * from customer_view;
+-----+-----+
c_address | c_name |
+-----+-----+
Farooqabad | NOOR |
Sheikhpura | NIMRA |
Sheikhpura | shanze |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

# LIKE CLAUSE ON CUSTOMER

```
MySQL 8.0 Command Line Client
1 row in set (0.00 sec)

mysql> select c_name from customer where c_name like 'S%';
+-----+
| c_name |
+-----+
| Shanzay |
+-----+
1 row in set (0.00 sec)

mysql> select c_name from customer where c_name like '%S';
+-----+
| c_name |
+-----+
| Noor |
+-----+
1 row in set (0.00 sec)

mysql> select c_name from customer where c_name like 'S%';
+-----+
| c_name |
+-----+
| Niera |
+-----+
1 row in set (0.00 sec)

mysql>
```

# JOINS

```
MySQL 8.0 Command Line Client - Unicode
Rows matched: 1  Changed: 1  Warnings: 0

mysql> describe room_booking;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| room_id | int | NO | PRI | NULL | |
| r_no | int | YES | | NULL | |
| check_in_date | date | YES | | NULL | |
| check_out_date | date | YES | | NULL | |
| weekend_pricing | int | YES | | NULL | |
| holiday_pricing | int | YES | | NULL | |
| early_bird_offers | varchar(29) | YES | | NULL | |
| STAFF_ID | int | YES | MUL | NULL | |
| custom_id | int | YES | MUL | NULL | |
+-----+-----+-----+-----+-----+-----+
0 rows in set (0.00 sec)

mysql> select * from room_booking;
+-----+-----+-----+-----+-----+-----+-----+-----+
| room_id | r_no | check_in_date | check_out_date | weekend_pricing | holiday_pricing | early_bird_offers | STAFF_ID | custom_id |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10 | 202 | 2025-01-05 | 2025-01-25 | 8000 | 9000 | 40% | 1 | 100 |
| 20 | 504 | 2025-02-04 | 2025-02-08 | 7000 | 8000 | 30% | 2 | 200 |
| 30 | 301 | 2025-07-04 | 2025-07-08 | 8000 | 12000 | 80% | 3 | 300 |
+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select room_booking.room_id,staff.s_name,customer.c_name from room_booking INNER JOIN staff ON room_booking.STAFF_ID = staff.s_id
INNER JOIN customer ON room_booking.custom_id=customer.c_id;
+-----+-----+-----+
| room_id | s_name | c_name |
+-----+-----+-----+
| 10 | haras | NOOR |
| 20 | ahmad | NIMRA |
| 30 | soha11 | shanzay |
+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

# INNER JOIN

```
MySQL 8.0 Command Line Client - Unicode
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| room_id | int | NO | PRI | NULL | |
| r_no | int | YES | | NULL | |
| check_in_date | date | YES | | NULL | |
| check_out_date | date | YES | | NULL | |
| weekend_pricing | int | YES | | NULL | |
| holiday_pricing | int | YES | | NULL | |
| early_bird_offers | varchar(29) | YES | | NULL | |
| STAFF_ID | int | YES | MUL | NULL | |
| custom_id | int | YES | MUL | NULL | |
+----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> describe customer;
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| c_id | int | NO | PRI | NULL | |
| c_name | varchar(15) | YES | | NULL | |
| r_id | int | YES | MUL | NULL | |
| c_address | varchar(30) | YES | | NULL | |
| phone_no | int | YES | | NULL | |
+----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select room_booking.r_no , staff.s_name from room_booking INNER join staff ON staff.s_id=room_booking.STAFF_ID;
+----+-----+
| r_no | s_name |
+----+-----+
| 202 | haran |
| 504 | ahmad |
| 301 | sohail |
+----+-----+
3 rows in set (0.05 sec)

mysql> _
```

```
MySQL 8.0 Command Line Client - Unicode
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right
syntax to use near '4, 'hamza', 'muree')' at line 1
mysql> INSERT INTO staff values(4, 'hamza', 'muree');
ERROR 1136 (21501): Column count doesn't match value count at row 1
mysql> INSERT INTO staff values(4, 'hamza', 'muree', 'housekeeping', 40, 15000);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails ('hotel_chain`.`staff`, CONSTRAINT `staff_ibfk_1`
FOREIGN KEY (`r_id`) REFERENCES `room_booking` (`room_id`))
mysql> INSERT INTO staff(s_id,s_name,s_address) values(4, 'hamza', 'muree');
Query OK, 1 row affected (0.02 sec)

mysql> select * from staff;
+----+-----+-----+-----+-----+
| s_id | s_name | s_address | s_role | r_id | salary |
+----+-----+-----+-----+-----+
| 1 | haran | multan | manager | 10 | 12000 |
| 2 | ahmad | sheikhpura | sweeper | 20 | 8000 |
| 3 | sohail | lahore | manager | 30 | 13000 |
| 4 | hamza | muree | NULL | NULL | NULL |
+----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select room_booking.r_no , staff.s_name from room_booking INNER join staff ON staff.s_id=room_booking.STAFF_ID;
+----+-----+
| r_no | s_name |
+----+-----+
| 202 | haran |
| 504 | ahmad |
| 301 | sohail |
+----+-----+
3 rows in set (0.00 sec)

mysql> _
```



# EQUI JOIN

```
MySQL 8.0 Command Line Client - Unicode
+----+-----+-----+-----+-----+
fb_id      int      NO      PRI      NULL
fb_type    varchar(20) YES     YES      NULL
comment    varchar(30) YES     YES      NULL
customer_id int      YES     MUL      NULL
+----+-----+-----+-----+
1 rows in set (0.00 sec)

mysql> describe customer;
+----+-----+-----+-----+-----+
Field      Type      Null      Key      Default      Extra
+----+-----+-----+-----+-----+
c_id       int      NO      PRI      NULL
c_name     varchar(15) YES     YES      NULL
r_id       int      YES     MUL      NULL
c_address  varchar(30) YES     YES      NULL
phone_no   int      YES     YES      NULL
+----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> select room_booking.weekend_pricing, room_booking.r_no, customer.c_name, phone_no from room_booking, customer where customer.c_id=room_booking.customer_id;
ERROR 1054 (42S22): unknown column 'room_booking.r_no' in 'field list'
mysql> select room_booking.weekend_pricing, room_booking.r_no, customer.c_name, phone_no from room_booking, customer where customer.c_id=room_booking.customer_id;
+-----+-----+-----+-----+
weekend_pricing | r_no | c_name | phone_no |
+-----+-----+-----+-----+
8000            | 202  | NOOR   | 38876257 |
7000            | 504  | NIMHA  | 38887257 |
8000            | 301  | shanzey | 333447257 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> -
```

# LEFT JOIN

```
MySQL 8.0 Command Line Client - Unicode
mysql> describe customer;
+----+-----+-----+-----+-----+
Field      Type      Null      Key      Default      Extra
+----+-----+-----+-----+-----+
c_id       int      NO      PRI      NULL
c_name     varchar(15) YES     YES      NULL
r_id       int      YES     MUL      NULL
c_address  varchar(30) YES     YES      NULL
phone_no   int      YES     YES      NULL
+----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> describe room_booking;
+----+-----+-----+-----+-----+
Field      Type      Null      Key      Default      Extra
+----+-----+-----+-----+-----+
room_id    int      NO      PRI      NULL
r_no       int      YES     YES      NULL
check_in_date date     YES     YES      NULL
check_out_date date     YES     YES      NULL
weekend_pricing int     YES     YES      NULL
holiday_pricing int     YES     YES      NULL
early_bird_offers varchar(29) YES     YES      NULL
STAFF_ID   int      YES     MUL      NULL
customer_id int      YES     MUL      NULL
+----+-----+-----+-----+-----+
9 rows in set (0.00 sec)

mysql> select room_booking.check_out_date, customer.c_address from room_booking LEFT JOIN customer ON customer.c_id = room_booking.customer_id;
+-----+-----+
check_out_date | c_address |
+-----+-----+
2025-01-25     | Farooqabad |
2025-02-08     | Sheikhpora |
2025-07-08     | Sheikhpora |
+-----+-----+
3 rows in set (0.00 sec)

mysql> -
```

## RIGHT JOIN

```

MySQL 8.0 Command Line Client - libcloud
+-----+-----+-----+-----+-----+-----+
| 20 | 504 | 2025-02-04 | 2025-02-08 | 7000 | 8000 | 30% | 2 | 200 | 300 |
| shanzey | 30 | sheikhpura | 333447257 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select room_booking.check_out_date,customer.c_address from room_booking RIGHT JOIN customer ON customer.c_id = room_booking.cust
om_id;
+-----+-----+
| check_out_date | c_address |
+-----+-----+
| 2025-01-25 | Farooqabad |
| 2025-02-08 | sheikhpura |
| 2025-07-08 | sheikhpura |
+-----+-----+
3 rows in set (0.00 sec)

mysql> select room_booking.check_out_date,customer.c_address from room_booking RIGHT JOIN customer ON customer.c_id = room_booking.cust
om_id;

```

## CROSS JOIN

```

MySQL 8.0 Command Line Client - Unicode
+-----+
| 30 | 301 | 2025-07-04 | 2025-07-08 | 8000 | 12000 | 80% | 3 | 300 |
| ahmad | sheikhpura | sweeper | 20 | 8000 | 300 | shanzey | 30 | sheikhpura | 333447257 | 1 | 100 |
| 10 | 202 | 2025-01-05 | 2025-01-25 | 8000 | 9000 | 40% | 1 | 100 |
| haran | multan | manager | 10 | 12000 | 300 | shanzey | 30 | sheikhpura | 333447257 | 1 | 100 |
| 20 | 504 | 2025-02-04 | 2025-02-08 | 7000 | 8000 | 30% | 2 | 200 |
| haran | multan | manager | 10 | 12000 | 300 | shanzey | 30 | sheikhpura | 333447257 | 1 | 100 |
| 30 | 301 | 2025-07-04 | 2025-07-08 | 8000 | 12000 | 80% | 3 | 300 |
| haran | multan | manager | 10 | 12000 | 300 | shanzey | 30 | sheikhpura | 333447257 | 1 | 100 |
+-----+
27 rows in set (0.03 sec)

mysql> select distinct staff_r_id,room_booking_r_no from staff cross join room_booking on staff_r_id=room_booking.room_id;
+-----+
| r_id | r_no |
+-----+
| 10 | 202 |
| 20 | 504 |
+-----+

```

| row_id | name     | email      | phone_area | phone_number | hire_date  | salary | commission_pct | manager_id | job_id | dept_name | emp_status | emp_type | emp_grade | emp_level | emp_status | emp_type | emp_grade | emp_level |
|--------|----------|------------|------------|--------------|------------|--------|----------------|------------|--------|-----------|------------|----------|-----------|-----------|------------|----------|-----------|-----------|
| 10     | John     | John.D     | 515        | 1212         | 1987-01-01 | 2800   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 11     | Paul     | Paul.H     | 515        | 1221         | 1987-02-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 12     | Henry    | Henry.K    | 515        | 1221         | 1987-03-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 13     | Albert   | Albert.E   | 515        | 1221         | 1987-04-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 14     | Mercedes | Mercedes.F | 515        | 1221         | 1987-05-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 15     | Pat      | Pat.S      | 515        | 1221         | 1987-06-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 16     | Neena    | Neena.K    | 515        | 1221         | 1987-07-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 17     | Lex      | Lex.D      | 515        | 1221         | 1987-08-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 18     | David    | David.A    | 515        | 1221         | 1987-09-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 19     | John     | John.D     | 515        | 1212         | 1987-01-01 | 2800   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 20     | Paul     | Paul.H     | 515        | 1221         | 1987-02-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 21     | Henry    | Henry.K    | 515        | 1221         | 1987-03-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 22     | Albert   | Albert.E   | 515        | 1221         | 1987-04-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 23     | Mercedes | Mercedes.F | 515        | 1221         | 1987-05-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 24     | Pat      | Pat.S      | 515        | 1221         | 1987-06-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 25     | Neena    | Neena.K    | 515        | 1221         | 1987-07-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 26     | Lex      | Lex.D      | 515        | 1221         | 1987-08-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 27     | David    | David.A    | 515        | 1221         | 1987-09-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 28     | John     | John.D     | 515        | 1212         | 1987-01-01 | 2800   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 29     | Paul     | Paul.H     | 515        | 1221         | 1987-02-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 30     | Henry    | Henry.K    | 515        | 1221         | 1987-03-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 31     | Albert   | Albert.E   | 515        | 1221         | 1987-04-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 32     | Mercedes | Mercedes.F | 515        | 1221         | 1987-05-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 33     | Pat      | Pat.S      | 515        | 1221         | 1987-06-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 34     | Neena    | Neena.K    | 515        | 1221         | 1987-07-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 35     | Lex      | Lex.D      | 515        | 1221         | 1987-08-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |
| 36     | David    | David.A    | 515        | 1221         | 1987-09-01 | 2000   | 0.1            | 1000       | SA     | SALES     | EMPLOYEE   | EMPLOYEE | SALES     | 10        | 1          | 1        | 1         | 1         |

# DELETE ROW4 FROM STAFF

```
MySQL 8.0 Command Line Client - Unicode
mysql> DELETE FROM staff WHERE s_role = 'sweeper';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails ('hotel_chain', 'hotel_branch', CONSTRAINT 'hotel_branch_ibfk_1' FOREIGN KEY (staff_id) REFERENCES staff (s_id))
mysql> select * from staff;
+----+-----+-----+-----+-----+-----+
| s_id | s_name | s_address | s_role | R_id | salary |
+----+-----+-----+-----+-----+-----+
| 1 | harge | multan | manager | 10 | 12000 |
| 2 | ahead | sheikhpura | sweeper | 20 | 8000 |
| 3 | sohail | lahore | manager | 30 | 13000 |
| 4 | hanza | muree | NULL | NULL | NULL |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> DELETE FROM staff WHERE s_address= 'muree';
Query OK, 1 row affected (0.01 sec)

mysql> select * from staff;
+----+-----+-----+-----+-----+-----+
| s_id | s_name | s_address | s_role | R_id | salary |
+----+-----+-----+-----+-----+-----+
| 1 | harge | multan | manager | 10 | 12000 |
| 2 | ahead | sheikhpura | sweeper | 20 | 8000 |
| 3 | sohail | lahore | manager | 30 | 13000 |
+----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

# DROP

```
mysql> drop task;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'task' at line 1
mysql> drop table task;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from task;
ERROR 1146 (42502): Table 'extra.task' doesn't exist
mysql> describe task;
ERROR 1146 (42502): Table 'extra.task' doesn't exist
mysql>
```

# TRUNCATE

```
MySQL 8.0 Command Line Client - Unicode
ERROR 1146 (42502): Table 'college.tables' doesn't exist
mysql> describe college;
ERROR 1146 (42502): Table 'college.college' doesn't exist
mysql> Drop table student;
Query OK, 0 rows affected (0.06 sec)

mysql> describe table;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'table' at line 1
mysql> show tables;
+-----+
| Tables_in_college |
+-----+
| course             |
+-----+
1 row in set (0.00 sec)

mysql> truncate table course;
Query OK, 0 rows affected (0.05 sec)

mysql> show tables;
+-----+
| Tables_in_college |
+-----+
| course             |
+-----+
1 row in set (0.00 sec)

mysql> select * from course;
Empty set (0.00 sec)

mysql> show database;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'database' at line 1
mysql> show databases;
+-----+
| Database |
+-----+
| college  |
| company |
| hotel_chain |
+-----+
3 rows in set (0.00 sec)
```

# SUBQUERY

```
MySQL 8.0 Command Line Client - Unicode
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 20 | 304 | 2025-02-04 | 2025-02-08 | 7000 | 8000 | 10% | 2 | 200 |
| 30 | 302 | 2025-07-04 | 2025-07-08 | 8000 | 12000 | 80% | 3 | 300 |
+----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 rows in set (0.04 sec)

mysql> select * from staff;
+----+-----+-----+-----+-----+-----+
| s_id | s_name | s_address | s_role | R_id | salary |
+----+-----+-----+-----+-----+-----+
| 1 | haran | multan | manager | 10 | 12000 |
| 2 | ahmad | sheikhpura | sweeper | 20 | 8000 |
| 3 | sohail | lahore | manager | 30 | 13000 |
+----+-----+-----+-----+-----+-----+
1 rows in set (0.03 sec)

mysql> select s_name, R_id from staff where salary=( select min(salary) from staff);
+-----+-----+
| s_name | R_id |
+-----+-----+
| ahmad | 20 |
+-----+-----+
1 row in set (0.01 sec)

mysql> select s_name, R_id from staff where salary=( select max(salary) from staff);
+-----+-----+
| s_name | R_id |
+-----+-----+
| sohail | 30 |
+-----+-----+
1 row in set (0.00 sec)

mysql> select s_name, R_id from staff where salary=( select avg(salary) from staff);
Empty set (0.00 sec)

mysql>
```

## LOWEST SALARY PRINTED

```
MySQL 8.0 Command Line Client - Unicode
+-----+-----+
| haran | 10 |
| sohail | 30 |
+-----+-----+
2 rows in set (0.00 sec)

mysql> select s_name, R_id , salary from staff where salary in( select min(salary) from staff group by s_role);
+-----+-----+-----+
| s_name | R_id | salary |
+-----+-----+-----+
| haran | 10 | 12000 |
| ahmad | 20 | 8000 |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

## MIN SALARY USING GROUP BY FROM S-ID PRINTED

```
mysql> select s_name, R_id, salary from staff where salary>any( select min(salary) from staff group by s_role);
+-----+-----+-----+
| s_name | R_id | salary |
+-----+-----+-----+
| haran | 10 | 12000 |
| sohail | 30 | 13000 |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select s_name, R_id, salary from staff where salary>ALL( select min(salary) from staff group by s_role);
+-----+-----+-----+
| s_name | R_id | salary |
+-----+-----+-----+
| sohail | 30 | 13000 |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```



# GRANT

```
MySQL 8.0 Command Line Client - Unicode

mysql> create user identified by '1234';
ERROR 1064 (42000): you have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right
syntax to use near 'by '1234'' at line 1
mysql> create user noor identified by '1234';
Query OK, 0 rows affected (0.06 sec)

mysql> create role manager;
ERROR 1396 (HY000): Operation CREATE ROLE failed for 'manager'@'%'
mysql> create role manager;
ERROR 1396 (HY000): Operation CREATE ROLE failed for 'manager'@'%'
mysql> create role manager_role;
Query OK, 0 rows affected (0.02 sec)

mysql> grant manager_role to noor;
Query OK, 0 rows affected (0.02 sec)

mysql> show grants;
+-----+
| Grants for root@localhost |
+-----+
|
+-----+
| GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER,
| CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER RO
| UTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE, CREATE ROLE, DROP ROLE ON *.* TO 'root'@'localhost' WITH GRANT OPTION
+-----+
```

```
MySQL 8.0 Command Line Client - Unicode

+-----+
| Grants for root@localhost |
+-----+
|
+-----+
| GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER,
| CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER RO
| UTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE, CREATE ROLE, DROP ROLE ON *.* TO 'root'@'localhost' WITH GRANT OPTION
+-----+
| GRANT APPLICATION_PASSWORD_ADMIN, AUDIT_ABORT_EXEMPT, AUDIT_ADMIN, AUTHENTICATION_POLICY_ADMIN, BACKUP_ADMIN, BINLOG_ADMIN, BINLOG_ENCRYPTI
| ON_ADMIN, CLONE_ADMIN, CONNECTION_ADMIN, ENCRYPTION_KEY_ADMIN, FIREWALL_EXEMPT, FLUSH_OPTIMIZER_COSTS, FLUSH_STATUS, FLUSH_TABLES, FLUSH_USER_R
| ESOURCES, GROUP_REPLICATION_ADMIN, GROUP_REPLICATION_STREAM, INNODB_REDO_LOG_ARCHIVE, INNODB_REDO_LOG_ENABLE, PASSWORDLESS_USER_ADMIN, PERST
| T_RO_VARIABLES_ADMIN, REPLICATION_APPLIER, REPLICATION_SLAVE_ADMIN, RESOURCE_GROUP_ADMIN, RESOURCE_GROUP_USER, ROLE_ADMIN, SENSITIVE_VARIABLE
| S_OBSERVER, SERVICE_CONNECTION_ADMIN, SESSION_VARIABLES_ADMIN, SET_USER_ID, SHOW_ROUTINE, SYSTEM_USER, SYSTEM_VARIABLES_ADMIN, TABLE_ENCRYPTIO
| N_ADMIN, TELEMETRY_LOG_ADMIN, XA_RECOVER_ADMIN ON *.* TO 'root'@'localhost' WITH GRANT OPTION |
+-----+
| GRANT PROXY ON '' TO 'root'@'localhost' WITH GRANT OPTION
+-----+

3 rows in set (0.01 sec)
```

# PRIVILEGES

MySQL 8.0 Command Line Client - Unicode

```
syntax to use near 'privileges' at line 1
mysql> show privileges;
```

| Privilege               | Context                                  | Comment                                               |
|-------------------------|------------------------------------------|-------------------------------------------------------|
| Alter                   | Tables                                   | To alter the table                                    |
| Alter routine           | Functions, Procedures                    | To alter or drop stored functions/procedures          |
| Create                  | Databases, Tables, Indexes               | To create new databases and tables                    |
| Create routine          | Databases                                | To use CREATE FUNCTION/PROCEDURE                      |
| Create role             | Server Admin                             | To create new roles                                   |
| Create temporary tables | Databases                                | To use CREATE TEMPORARY TABLE                         |
| Create view             | Tables                                   | To create new views                                   |
| Create user             | Server Admin                             | To create new users                                   |
| Delete                  | Tables                                   | To delete existing rows                               |
| Drop                    | Databases, Tables                        | To drop databases, tables, and views                  |
| Drop role               | Server Admin                             | To drop roles                                         |
| Event                   | Server Admin                             | To create, alter, drop and execute events             |
| Execute                 | Functions, Procedures                    | To execute stored routines                            |
| File                    | File access on server                    | To read and write files on the server                 |
| Grant option            | Databases, Tables, Functions, Procedures | To give to other users those privileges you possess   |
| Index                   | Tables                                   | To create or drop indexes                             |
| Insert                  | Tables                                   | To insert data into tables                            |
| Lock tables             | Databases                                | To use LOCK TABLES (together with SELECT privilege)   |
| Process                 | Server Admin                             | To view the plain text of currently executing queries |
| Proxy                   | Server Admin                             | To make proxy user possible                           |
| References              | Databases, Tables                        | To have references on tables                          |
| Reload                  | Server Admin                             | To reload or refresh tables, logs and privileges      |
| replication client      | Server Admin                             | To ask where the slave or master servers are          |
| replication slave       | Server Admin                             | To read binary log events from the master             |
| Select                  | Tables                                   | To retrieve rows from table                           |
| Show databases          | Server Admin                             | To see all databases with SHOW DATABASES              |
| Show view               | Tables                                   | To see views with SHOW CREATE VIEW                    |
| shutdown                | Server Admin                             | To shut down the server                               |
| Super                   | Server Admin                             | To use KILL thread, SET GLOBAL, CHANGE MASTER, etc.   |
| Trigger                 | Tables                                   | To use triggers                                       |
| Create tablespace       | Server Admin                             | To create/alter/drop tablespaces                      |
| update                  | Tables                                   | To update existing rows                               |
| Usage                   | Server Admin                             | No privileges - allow connect only                    |
| ENCRYPTION_KEY_ADMIN    | Server Admin                             |                                                       |
| INNODB_REDO_LOG_ARCHIVE | Server Admin                             |                                                       |
| RESOURCE_GROUP_USER     | Server Admin                             |                                                       |

MySQL 8.0 Command Line Client - Unicode

|                              |              |                                                     |
|------------------------------|--------------|-----------------------------------------------------|
| Shutdown                     | Server Admin | To shut down the server                             |
| Super                        | Server Admin | To use KILL thread, SET GLOBAL, CHANGE MASTER, etc. |
| Trigger                      | Tables       | To use triggers                                     |
| Create tablespace            | Server Admin | To create/alter/drop tablespaces                    |
| Update                       | Tables       | To update existing rows                             |
| Usage                        | Server Admin | No privileges - allow connect only                  |
| ENCRYPTION_KEY_ADMIN         | Server Admin |                                                     |
| INNODB_REDO_LOG_ARCHIVE      | Server Admin |                                                     |
| RESOURCE_GROUP_USER          | Server Admin |                                                     |
| FIREWALL_EXEMPT              | Server Admin |                                                     |
| SET_USER_ID                  | Server Admin |                                                     |
| SERVICE_CONNECTION_ADMIN     | Server Admin |                                                     |
| GROUP_REPLICATION_ADMIN      | Server Admin |                                                     |
| AUDIT_ABORT_EXEMPT           | Server Admin |                                                     |
| GROUP_REPLICATION_STREAM     | Server Admin |                                                     |
| CLONE_ADMIN                  | Server Admin |                                                     |
| SYSTEM_USER                  | Server Admin |                                                     |
| AUTHENTICATION_POLICY_ADMIN  | Server Admin |                                                     |
| SHOW_ROUTINE                 | Server Admin |                                                     |
| BACKUP_ADMIN                 | Server Admin |                                                     |
| CONNECTION_ADMIN             | Server Admin |                                                     |
| PERSIST_RO_VARIABLES_ADMIN   | Server Admin |                                                     |
| RESOURCE_GROUP_ADMIN         | Server Admin |                                                     |
| SESSION_VARIABLES_ADMIN      | Server Admin |                                                     |
| SYSTEM_VARIABLES_ADMIN       | Server Admin |                                                     |
| APPLICATION_PASSWORD_ADMIN   | Server Admin |                                                     |
| FLUSH_OPTIMIZER_COSTS        | Server Admin |                                                     |
| AUDIT_ADMIN                  | Server Admin |                                                     |
| BINLOG_ADMIN                 | Server Admin |                                                     |
| BINLOG_ENCRYPTION_ADMIN      | Server Admin |                                                     |
| FLUSH_STATUS                 | Server Admin |                                                     |
| FLUSH_TABLES                 | Server Admin |                                                     |
| FLUSH_USER_RESOURCES         | Server Admin |                                                     |
| REPLICATION_APPLIER          | Server Admin |                                                     |
| INNODB_REDO_LOG_ENABLE       | Server Admin |                                                     |
| XA_RECOVER_ADMIN             | Server Admin |                                                     |
| PASSWORDLESS_USER_ADMIN      | Server Admin |                                                     |
| TABLE_ENCRYPTION_ADMIN       | Server Admin |                                                     |
| ROLE_ADMIN                   | Server Admin |                                                     |
| REPLICATION_SLAVE_ADMIN      | Server Admin |                                                     |
| SENSITIVE_VARIABLES_OBSERVER | Server Admin |                                                     |

```
MySQL 8.0 Command Line Client - Unicode
update
usage
Server Admin
ENCRYPTION_KEY_ADMIN
Server Admin
INNODB_REDO_LOG_ARCHIVE
Server Admin
RESOURCE_GROUP_USER
Server Admin
FIREWALL_EXEMPT
Server Admin
SET_USER_ID
Server Admin
SERVICE_CONNECTION_ADMIN
Server Admin
GROUP_REPLICATION_ADMIN
Server Admin
AUDIT_ABORT_EXEMPT
Server Admin
GROUP_REPLICATION_STREAM
Server Admin
CLONE_ADMIN
Server Admin
SYSTEM_USER
Server Admin
AUTHENTICATION_POLICY_ADMIN
Server Admin
SHOW_ROUTINE
Server Admin
BACKUP_ADMIN
Server Admin
CONNECTION_ADMIN
Server Admin
PERSIST_BO_VARIABLES_ADMIN
Server Admin
RESOURCE_GROUP_ADMIN
Server Admin
SESSION_VARIABLES_ADMIN
Server Admin
SYSTEM_VARIABLES_ADMIN
Server Admin
APPLICATION_PASSWORD_ADMIN
Server Admin
FLUSH_OPTIMIZER_COSTS
Server Admin
AUDIT_ADMIN
Server Admin
BINLOG_ADMIN
Server Admin
BINLOG_ENCRYPTION_ADMIN
Server Admin
FLUSH_STATUS
Server Admin
FLUSH_TABLES
Server Admin
FLUSH_USER_RESOURCES
Server Admin
REPLICATION_APPLIER
Server Admin
INNODB_REDO_LOG_ENABLE
Server Admin
XA_RECOVER_ADMIN
Server Admin
PASSWORDLESS_USER_ADMIN
Server Admin
TABLE_ENCRYPTION_ADMIN
Server Admin
ROLE_ADMIN
Server Admin
REPLICATION_SLAVE_ADMIN
Server Admin
SENSITIVE_VARIABLES_OBSERVER
Server Admin
TELEMETRY_LOG_ADMIN
Server Admin
Tables
To update existing rows
No privileges - allow connect only
69 rows in set (0.01 sec)
```