



SQL COMMANDS

IMPLEMENTATION OF COMMANDS IN SQL
ELECTRONIC PRODUCT SERVICE SYSTEM

Eman Dar

BACHELOR OF INFORMATION TECHNOLOGY | GOVT GRADUATE
COLLEGE FOR BOYS CIVIL LINES, SHEIKHUPPURA

SQL COMMANDS

- CREATE DATABASE
- DESCRIBE TABLE (DESC)
- INSERT INTO
- ADD VALUES IN TABLES
- ADD FOREIGN KEY OR MULTIPLE FOREIGN KEY
- SELECT COMMAND
- UPDATE
- AS
- DSITINCT
- ORDER BY
- WHERE clause
- SELECT COMMAND
- ARITHMETIC OPERATORS
- RELATIONAL OPERATORS
- LOGICAL OPERATORS
- AND BETWEEN
- IN
- LIKE
- NULL and NOT NULL
- AGREGRATE FUNCTIONS
- GROUP BY
- HAVING
- JOIN
 - INNER
 - OUTER
 - RIGHT
 - LEFT
- DELETE TABLE / DATABASE

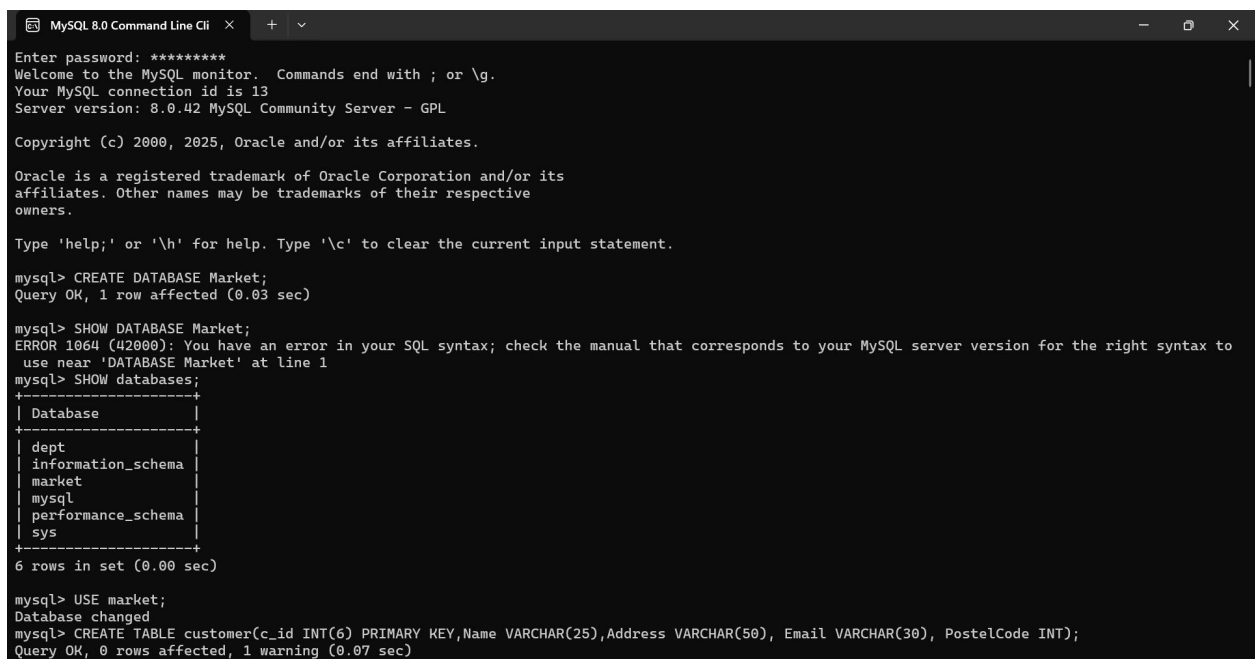
IMPLEMENTATIONS OF DIFFERENT COMMANDS ON MYSQL

CREATE DATABASE

Procedure to create Database in SQL command Line:

1. Open the SQL Command Line Client.
2. Enter the password.
3. Enter the command to show the databases→ **SHOW DATABASES;**
4. If database is already existing then use command to manipulate in target database.

- **USE database_name;**
- Database name: market;
- Example: **use market;**



```
MySQL 8.0 Command Line Cli
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 13
Server version: 8.0.42 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

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 Market;
Query OK, 1 row affected (0.03 sec)

mysql> SHOW DATABASE Market;
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 Market' at line 1
mysql> SHOW databases;
+-----+
| Database |
+-----+
| dept     |
| information_schema |
| market   |
| mysql    |
| performance_schema |
| sys      |
+-----+
6 rows in set (0.00 sec)

mysql> USE market;
Database changed
mysql> CREATE TABLE customer(c_id INT(6) PRIMARY KEY,Name VARCHAR(25),Address VARCHAR(50), Email VARCHAR(30), PostelCode INT);
Query OK, 0 rows affected, 1 warning (0.07 sec)
```

- **Show tables;**
- If you have just created your database so create tables as; **CREATE TABLE table_name();**
- **CREATE TABLE customer(c_id INT(6) PRIMARY KEY,Name VARCHAR(25),Address VARCHAR(50), Email VARCHAR(30), PostelCode INT);**
- Check the description of the table as; **DESC table_name;**
- **DESCRIBE customer;**

```

MySQL 8.0 Command Line Cli x + v
Database changed
mysql> CREATE TABLE customer(c_id INT(6) PRIMARY KEY,Name VARCHAR(25),Address VARCHAR(50), Email VARCHAR(30), PostalCode INT);
Query OK, 0 rows affected, 1 warning (0.07 sec)

mysql> DESCRIBE customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| c_id  | int  | NO   | PRI | NULL    |       |
| Name  | varchar(25) | YES |     | NULL    |       |
| Address | varchar(50) | YES |     | NULL    |       |
| Email  | varchar(30) | YES |     | NULL    |       |
| PostalCode | int  | YES |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.04 sec)

mysql> CREATE TABLE product(
  -> product_id INT(6) PRIMARY KEY,
  -> Model_no VARCHAR(50),
  -> Product_type VARCHAR(50),
  -> Price DECIMAL(10,2),
  -> Serial_no VARCHAR(50)
  -> );
Query OK, 0 rows affected, 1 warning (0.05 sec)

mysql> DESCRIBE product;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| product_id | int  | NO   | PRI | NULL    |       |
| Model_no   | varchar(50) | YES |     | NULL    |       |
| Product_type | varchar(50) | YES |     | NULL    |       |
| Price      | decimal(10,2) | YES |     | NULL    |       |
| Serial_no  | varchar(50) | YES |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

```

- Also create other tables using same command.
- Technician and manufacturer table;

```

MySQL 8.0 Command Line Cli x + v

mysql> DESCRIBE technician;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| technician_id | int  | NO   | PRI | NULL    |       |
| Name          | varchar(25) | YES |     | NULL    |       |
| Expertise     | varchar(50) | YES |     | NULL    |       |
| Contact_info  | varchar(70) | YES |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> CREATE TABLE manufacturer(manufacturer_id INT PRIMARY KEY, Name
  -> VARCHAR(25), Address Va
  -> VARCHAR(25), Address Va
  -> Contact_info VARCHAR(50));
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 'Va
VARCHAR(25), Address Va
Contact_info VARCHAR(50))' at line 2
mysql> CREATE TABLE manufacturer(manufacturer_id INT PRIMARY KEY, Name VARCHAR(25), Address VARCHAR(50), Contact_info VARCHAR(50));
Query OK, 0 rows affected (0.04 sec)

mysql> DESCRIBE manufacturer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| manufacturer_id | int  | NO   | PRI | NULL    |       |
| Name            | varchar(25) | YES |     | NULL    |       |
| Address         | varchar(50) | YES |     | NULL    |       |
| Contact_info    | varchar(50) | YES |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

- Warranty table;

```

MySQL 8.0 Command Line Cli x + v

mysql> desc warranty;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| warranty_id | int  | NO   | PRI | NULL    |       |
| warranty_type | varchar(50) | YES |     | NULL    |       |
| coverage_details | varchar(50) | YES |     | NULL    |       |
| product_id  | int  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

- Service request table;
- Alter the table attributes by using the command; **ALTER TABLE servicerequest ADD c_id int(6);**

```
mysql> CREATE TABLE servicerequest(service_req_id INT PRIMARY KEY,issue_desc VARCHAR(100),Request_date DATE);
Query OK, 0 rows affected (0.05 sec)

mysql> ALTER TABLE servicerequest ADD product_id INT(6), c_id INT(6);
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 'c_id INT(6)' at line 1
mysql> ALTER TABLE servicerequest ADD product_id INT(6);
Query OK, 0 rows affected, 1 warning (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> ALTER TABLE servicerequest ADD c_id int(6);
Query OK, 0 rows affected, 1 warning (0.07 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> DESC servicerequest;
```

Field	Type	Null	Key	Default	Extra
service_req_id	int	NO	PRI	NULL	
issue_desc	varchar(100)	YES		NULL	
Request_date	date	YES		NULL	
product_id	int	YES		NULL	
c_id	int	YES		NULL	

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

- Repair history table;

```
mysql> desc repairhistory;
```

Field	Type	Null	Key	Default	Extra
repair_history_id	int	NO	PRI	NULL	
Repair_date	date	YES		NULL	
Description	varchar(100)	YES		NULL	
cost	decimal(10,2)	YES		NULL	

```
4 rows in set (0.00 sec)

mysql> ALTER TABLE servicerequest
->
->
-> ALTER TABLE repairhistory ADD service_req_id int(6);
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 repairhistory ADD service_req_id int(6)' at line 5
mysql> ALTER TABLE repairhistory ADD service_req_id int(6);
Query OK, 0 rows affected, 1 warning (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> ALTER TABLE repairhistory ADD technician_id int(6);
Query OK, 0 rows affected, 1 warning (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> desc repairhistory;
```

Field	Type	Null	Key	Default	Extra
repair_history_id	int	NO	PRI	NULL	
Repair_date	date	YES		NULL	
Description	varchar(100)	YES		NULL	
cost	decimal(10,2)	YES		NULL	
service_req_id	int	YES		NULL	
technician_id	int	YES		NULL	

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

- Replacement part used table;

```
mysql> CREATE TABLE replacementpartused (rpu_id int PRIMARY KEY , repair_history_id int, quantity int);
Query OK, 0 rows affected (0.04 sec)

mysql> desc replacementpartused;
```

Field	Type	Null	Key	Default	Extra
rpu_id	int	NO	PRI	NULL	
repair_history_id	int	YES		NULL	
quantity	int	YES		NULL	

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

CREATE TABLES

It's used for the creation of tables. Evolving the following statement;

Syntax:

CREATE TABLE table_name (column_1,column_2,.....);

- CREATE TABLE **customer**(c_id INT(6) PRIMARY KEY,Name VARCHAR(25),Address VARCHAR(50), Email VARCHAR(30), PostalCode INT);
- Check the description of the table as; DESC table_name;
- DESCRIBE customer;

```
MySQL 8.0 Command Line Cli
Database changed
mysql> CREATE TABLE customer(c_id INT(6) PRIMARY KEY,Name VARCHAR(25),Address VARCHAR(50), Email VARCHAR(30), PostalCode INT);
Query OK, 0 rows affected, 1 warning (0.07 sec)

mysql> DESCRIBE customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| c_id  | int  | NO   | PRI | NULL    |       |
| Name  | varchar(25) | YES |     | NULL    |       |
| Address | varchar(50) | YES |     | NULL    |       |
| Email  | varchar(30) | YES |     | NULL    |       |
| PostalCode | int  | YES |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.04 sec)

mysql> CREATE TABLE product(
-> product_id INT(6) PRIMARY KEY,
-> Model_no VARCHAR(50),
-> Product_type VARCHAR(50),
-> Price DECIMAL(10,2),
-> Serial_no VARCHAR(50)
-> );
Query OK, 0 rows affected, 1 warning (0.05 sec)

mysql> DESCRIBE product;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| product_id | int | NO | PRI | NULL |       |
| Model_no   | varchar(50) | YES |     | NULL |       |
| Product_type | varchar(50) | YES |     | NULL |       |
| Price      | decimal(10,2) | YES |     | NULL |       |
| Serial_no  | varchar(50) | YES |     | NULL |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Use of DESCRIBE COMMAND

It's also can be used as the DESC shortly.

Syntax:

DESC table_name;

```
MySQL 8.0 Command Line Cli
mysql> desc customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| c_id  | int  | NO   | PRI | NULL    |       |
| Name  | varchar(25) | YES |     | NULL    |       |
| Address | varchar(50) | YES |     | NULL    |       |
| Email  | varchar(30) | YES |     | NULL    |       |
| PostalCode | int  | YES |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)
```

INSERT COMMAD

Syntax;

INSERT INTO table_name VALUES (column_1 , VALUES_C.....);

- Insert values in tables by using command as;
INSERT INTO manufacturer values
('101', 'Ali', 'skp', '0321-1234567'),
('102', 'Alyan', 'lhr', '0321-1234569'),
('103', 'Arman', 'fsd', '0321-1234568');

```
MySQL 8.0 Command Line Cli
4 rows in set (0.00 sec)
mysql> INSERT INTO manufacturer values('101','Ali','skp','0321-1234567'),('101','Alyan','lhr','0321-1234569'),('103','Arman','fsd','0321-1234568');
ERROR 1062 (23000): Duplicate entry '101' for key 'manufacturer.PRIMARY'
mysql> INSERT INTO manufacturer values('101','Ali','skp','0321-1234567'),('102','Alyan','lhr','0321-1234569'),('103','Arman','fsd','0321-1234568');
Query OK, 3 rows affected (0.02 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> DESC manufacturer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| manufacturer_id | int | NO | PRI | NULL | |
| Name | varchar(25) | YES | | NULL | |
| Address | varchar(50) | YES | | NULL | |
| Contact_info | varchar(50) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select *from manufacturer;
+-----+-----+-----+-----+
| manufacturer_id | Name | Address | Contact_info |
+-----+-----+-----+-----+
| 101 | Ali | skp | 0321-1234567 |
| 102 | Alyan | lhr | 0321-1234569 |
| 103 | Arman | fsd | 0321-1234568 |
+-----+-----+-----+-----+
3 rows in set (0.02 sec)

mysql> desc customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| c_id | int | NO | PRI | NULL | |
| Name | varchar(25) | YES | | NULL | |
| Address | varchar(50) | YES | | NULL | |
| Email | varchar(30) | YES | | NULL | |
| PostalCode | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

INSERT INTO customer VALUES('1','Azka','fsd','azka89@gmail.com','345'),('2','zayan','skp','zayan80@gmail.com','567'),('3','Anaiza','lhr','anaiza77@gmail.com','897');
Query OK, 3 rows affected (0.03 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from customer;
+-----+-----+-----+-----+-----+
| c_id | Name | Address | Email | PostalCode |
+-----+-----+-----+-----+-----+
| 1 | Azka | fsd | azka89@gmail.com | 345 |
| 2 | zayan | skp | zayan80@gmail.com | 567 |
| 3 | Anaiza | lhr | anaiza77@gmail.com | 897 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
|  |  |  |  |  |  |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

ADD Values in tables

- Insert values in tables by using command as;
INSERT INTO manufacturer values
('101', 'Ali', 'skp', '0321-1234567'),
('102', 'Alyan', 'lhr', '0321-1234569'),
('103', 'Arman', 'fsd', '0321-1234568');

```

MySQL 8.0 Command Line Cli
4 rows in set (0.00 sec)

mysql> INSERT INTO manufacturer values('101','Ali','skp','0321-1234567'),('101','Alyan','lhr','0321-1234569'),('103','Arman','fsd','0321-1234568');
ERROR 1062 (23000): Duplicate entry '101' for key 'manufacturer.PRIMARY'
mysql> INSERT INTO manufacturer values('101','Ali','skp','0321-1234567'),('102','Alyan','lhr','0321-1234569'),('103','Arman','fsd','0321-1234568');
Query OK, 3 rows affected (0.02 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> DESC manufacturer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| manufacturer_id | int | NO | PRI | NULL | |
| Name | varchar(25) | YES | | NULL | |
| Address | varchar(50) | YES | | NULL | |
| Contact_info | varchar(50) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select *from manufacturer;
+-----+-----+-----+-----+
| manufacturer_id | Name | Address | Contact_info |
+-----+-----+-----+-----+
| 101 | Ali | skp | 0321-1234567 |
| 102 | Alyan | lhr | 0321-1234569 |
| 103 | Arman | fsd | 0321-1234568 |
+-----+-----+-----+-----+
3 rows in set (0.02 sec)

mysql> desc customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| c_id | int | NO | PRI | NULL | |
| Name | varchar(25) | YES | | NULL | |
| Address | varchar(50) | YES | | NULL | |
| Email | varchar(30) | YES | | NULL | |
| PostalCode | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> INSERT INTO customer VALUES('1','Azka','fqd','azka09@gmail.com','345'),('2','zayan','skp','zayan08@gmail.com','567'),('3','Anaiza','lhr','anaiza77@gmail.com','897');
Query OK, 3 rows affected (0.03 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from customer;
+-----+-----+-----+-----+-----+
| c_id | Name | Address | Email | PostalCode |
+-----+-----+-----+-----+-----+
| 1 | Azka | fqd | azka09@gmail.com | 345 |
| 2 | zayan | skp | zayan08@gmail.com | 567 |
| 3 | Anaiza | lhr | anaiza77@gmail.com | 897 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+

```

Same as;

```

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

mysql> desc servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| service_req_id | int | NO | PRI | NULL | |
| issue_desc | varchar(100) | YES | | NULL | |
| Request_date | date | YES | | NULL | |
| product_id | int | YES | MUL | NULL | |
| c_id | int | YES | MUL | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> INSERT INTO servicerequest VALUES('401','Resolved','2024-11-12','11','screen not turning on'),('402','In progress','2024-01-18','12','motor making noise'),('403','May take time','2024-03-14','13','technical faults');
ERROR 1366 (HY000): Incorrect integer value: 'screen not turning on' for column 'c_id' at row 1
mysql> INSERT INTO servicerequest VALUES('401','Resolved','2024-11-12','11','1'),('402','In progress','2024-01-18','12','2'),('403','May take time','2024-03-14','13','3');
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails ('market'.servicerequest, CONSTRAINT 'fk_servicerequest' FOREIGN KEY ('product_id') REFERENCES 'product' ('product_id'))
mysql> desc product;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| product_id | int | NO | PRI | NULL | |
| Model_no | varchar(50) | YES | | NULL | |
| Product_type | varchar(50) | YES | | NULL | |
| Price | decimal(10,2) | YES | | NULL | |
| Serial_no | varchar(50) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> INSERT INTO product VALUES ('301','MOD-A1','Smartphone','899.99','SN-XYZ001'),('302','MOD-B2','Laptop','1499.50','SN-ABC202'),('303','MOD-C3','Headphones','199.00','SN-QWE345');
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from product;
+-----+-----+-----+-----+-----+
| product_id | Model_no | Product_type | Price | Serial_no |
+-----+-----+-----+-----+-----+
| 301 | MOD-A1 | Smartphone | 899.99 | SN-XYZ001 |
| 302 | MOD-B2 | Laptop | 1499.50 | SN-ABC202 |
| 303 | MOD-C3 | Headphones | 199.00 | SN-QWE345 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| service_req_id | int | NO | PRI | NULL | |
| issue_desc | varchar(100) | YES | | NULL | |
| Request_date | date | YES | | NULL | |
| product_id | int | YES | MUL | NULL | |
| c_id | int | YES | MUL | NULL | |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql> INSERT INTO servicerequest VALUES('401','Resolved','2024-11-12','301','1'),('402','In progress','2024-01-18','302','2'),('403','May take time','2024-03-14','303','3');

```

Also same for others;


```

MySQL 8.0 Command Line Cli
5 rows in set (0.00 sec)

mysql> INSERT INTO servicerequest VALUES('401','Resolved','2024-11-12','301','1'),('402','In progress','2024-01-18','302','2'),('403','May take time','2024-03-14','303','3');
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from servicerequest;
+-----+-----+-----+-----+-----+
| service_req_id | issue_desc | Request_date | product_id | c_id |
+-----+-----+-----+-----+-----+
| 401 | Resolved | 2024-11-12 | 301 | 1 |
| 402 | In progress | 2024-01-18 | 302 | 2 |
| 403 | May take time | 2024-03-14 | 303 | 3 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc replacementpartused;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| rpu_id | int | NO | PRI | NULL | |
| repair_history_id | int | YES | MUL | NULL | |
| quantity | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc repairhistory;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| repair_history_id | int | NO | PRI | NULL | |
| Repair_date | date | YES | | NULL | |
| Description | varchar(100) | YES | | NULL | |
| cost | decimal(10,2) | YES | | NULL | |
| service_req_id | int | YES | MUL | NULL | |
| technician_id | int | YES | MUL | NULL | |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)

mysql> desc technician;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| technician_id | int | NO | PRI | NULL | |
| Name | varchar(25) | YES | | NULL | |
| Expertise | varchar(50) | YES | | NULL | |
| Contact_info | varchar(70) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> INSERT INTO technician VALUES ('600','Liam O'Connor','Aerospace Avionics Repair','liam.avionics@skytech.ie | +353 87 123 4567'),('605','Sofia Rossi','Vintage Watch Restoration','sofia.restoration@horology.it (Turin Works hoo!)'),('606','Amir Al-Farsi','Oil Rig Industrial Sensors','+966 55 678 9012 | Emergency Line: 1990');
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from technician;
+-----+-----+-----+-----+
| technician_id | Name | Expertise | Contact_info |
+-----+-----+-----+-----+
| 600 | Liam O'Connor | Aerospace Avionics Repair | liam.avionics@skytech.ie | +353 87 123 4567 |
| 605 | Sofia Rossi | Vintage Watch Restoration | sofia.restoration@horology.it (Turin Workshop) |
| 606 | Amir Al-Farsi | Oil Rig Industrial Sensors | +966 55 678 9012 | Emergency Line: 1990 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

```

```

MySQL 8.0 Command Line Cli

mysql> select *from technician;
+-----+-----+-----+-----+
| technician_id | Name | Expertise | Contact_info |
+-----+-----+-----+-----+
| 600 | Liam O'Connor | Aerospace Avionics Repair | liam.avionics@skytech.ie | +353 87 123 4567 |
| 605 | Sofia Rossi | Vintage Watch Restoration | sofia.restoration@horology.it (Turin Workshop) |
| 606 | Amir Al-Farsi | Oil Rig Industrial Sensors | +966 55 678 9012 | Emergency Line: 1990 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> INSERT INTO technicians
-> VALUES
-> ('404','Liam O'Connor','Aerospace Avionics Repair','liam.avionics@skytech.ie | +353 87 123 4567'),
-> ('505','Sofia Rossi','Vintage Watch Restoration','sofia.restoration@horology.it (Turin Workshop)'),
-> ('606','Amir Al-Farsi','Oil Rig Industrial Sensors','+966 55 678 9012 | Emergency Line: 1990');
ERROR 1146 (42S02): Table 'market.technicians' doesn't exist
mysql> INSERT INTO repairhistory VALUES ('1001','2025-05-15','Screen replacement for device X203','89.99','401','604'),('1002','2025-05-16','Battery calibration and optimization','45.50','402','605'),('1003','2025-04-28','Motherboard diagnostics and component replacement','299.00','401','606');
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 'INSERT INTO repairhistory VALUES ('1001','2025-05-15','Screen replacement for d' at line 1
mysql> INSERT INTO repairhistory VALUES ('1001','2025-05-15','Screen replacement for device X203','89.99','401','604'),('1002','2025-05-16','Battery calibration and optimization','45.50','402','605'),('1003','2025-04-28','Motherboard diagnostics and component replacement','299.00','401','606');
Query OK, 3 rows affected (0.03 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from repairhistory;
+-----+-----+-----+-----+-----+-----+
| repair_history_id | Repair_date | Description | cost | service_req_id | technician_id |
+-----+-----+-----+-----+-----+-----+
| 1001 | 2025-05-15 | Screen replacement for device X203 | 89.99 | 401 | 604 |
| 1002 | 2025-05-16 | Battery calibration and optimization | 45.50 | 402 | 605 |
| 1003 | 2025-04-28 | Motherboard diagnostics and component replacement | 299.00 | 401 | 606 |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc replacementpartused;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| rpu_id | int | NO | PRI | NULL | |
| repair_history_id | int | YES | MUL | NULL | |
| quantity | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> INSERT INTO replacementpartused VALUES ('11','1001','10'),('12','1002','20'),('13','1003','30');
Query OK, 3 rows affected (0.02 sec)
Records: 3 Duplicates: 0 Warnings: 0

mysql> select *from replacementpartused;
+-----+-----+-----+
| rpu_id | repair_history_id | quantity |
+-----+-----+-----+
| 11 | 1001 | 10 |
| 12 | 1002 | 20 |
| 13 | 1003 | 30 |
+-----+-----+-----+
3 rows in set (0.00 sec)

```

ADD FOREIGN KEY

First of all we add the attribute in the child table as an alias and then reference to the parent PK.

We want to add the FK in the in the student table from course table here student table is child and course table is parent table. First of all we add the attribute as child in the student table and then reference it.

- Use the following command to add new attribute to the table.

Command: **ALTER TABLE servicerequest ADD c_id int(6);**

- Check the service request table ;

```
MySQL 8.0 Command Line Cli
Records: 0 Duplicates: 0 Warnings: 1

mysql> DESC servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| service_req_id | int       | NO   | PRI | NULL    |       |
| issue_desc    | varchar(100) | YES  |     | NULL    |       |
| Request_date  | date      | YES  |     | NULL    |       |
| product_id    | int       | YES  |     | NULL    |       |
| c_id         | int       | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

- Now make it as FK using following commands.
- Command : `ALTER TABLE servicerequest ADD CONSTRAINT fk_servicerequest FOREIGN KEY (product_id) REFERENCES product(product_id);`

Or

`ALTER TABLE servicerequest ADD CONSTRAINT fk_servicerequest FOREIGN KEY (c_id) REFERENCES customer(c_id);`

```
mysql> ALTER TABLE servicerequest ADD CONSTRAINT fk_servicerequest FOREIGN KEY (product_id) REFERENCES product(product_id);
Query OK, 0 rows affected (0.10 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> ALTER TABLE servicerequest ADD CONSTRAINT fk_servicerequest FOREIGN KEY (c_id) REFERENCES customer(c_id);
ERROR 1061 (42000): Duplicate key name 'fk_servicerequest'
mysql> desc servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| service_req_id | int       | NO   | PRI | NULL    |       |
| issue_desc    | varchar(100) | YES  |     | NULL    |       |
| Request_date  | date      | YES  |     | NULL    |       |
| product_id    | int       | YES  | MUL | NULL    |       |
| c_id         | int       | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

- Now for two foreign keys in same table use the command as;

`ALTER TABLE servicerequest`

- > `ADD CONSTRAINT fk_servicerequest_customer2`
- > `FOREIGN KEY (c_id)`
- > `REFERENCES customer(c_id)`
- > `ON DELETE CASCADE;`

```
mysql> ALTER TABLE servicerequest
-> ADD CONSTRAINT fk_servicerequest_customer2 -- Unique constraint name
-> FOREIGN KEY (c_id)
-> REFERENCES customer(c_id)
-> ON DELETE CASCADE; -- Optional: Add referential action
Query OK, 0 rows affected (0.11 sec)
Records: 0 Duplicates: 0 Warnings: 0

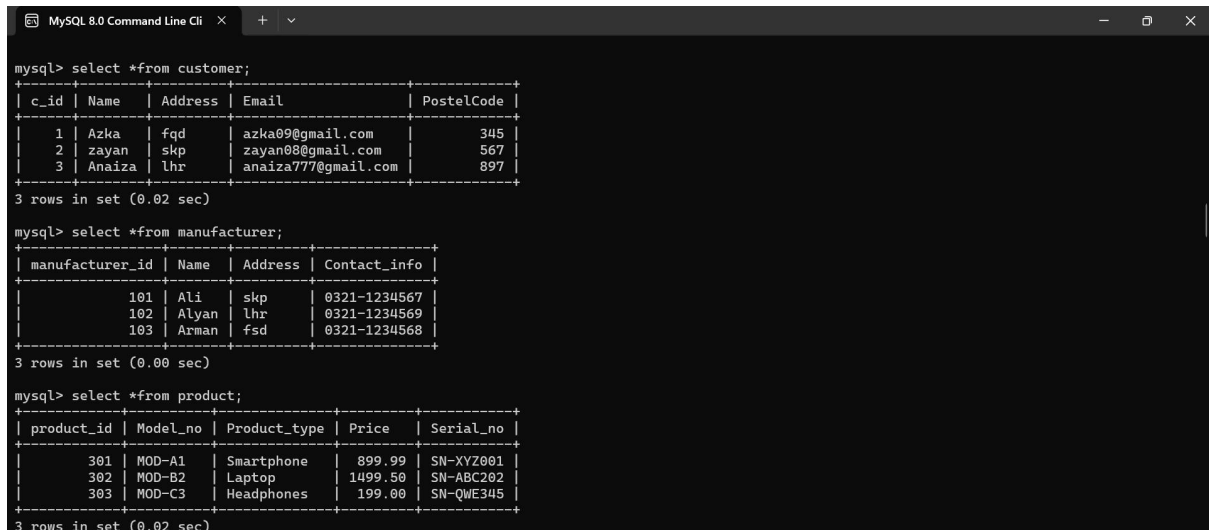
mysql> desc servicerequest;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| service_req_id | int       | NO   | PRI | NULL    |       |
| issue_desc    | varchar(100) | YES  |     | NULL    |       |
| Request_date  | date      | YES  |     | NULL    |       |
| product_id    | int       | YES  | MUL | NULL    |       |
| c_id         | int       | YES  | MUL | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

SELECT * command

This command is used for the displaying all and every single attribute's values on the command prompt.

Syntax;

SELECT *FROM table_name;



```
mysql> select *from customer;
+-----+-----+-----+-----+-----+
| c_id | Name | Address | Email | PostalCode |
+-----+-----+-----+-----+-----+
| 1 | Azka | fqd | azka09@gmail.com | 345 |
| 2 | zayan | skp | zayan08@gmail.com | 567 |
| 3 | Anaiza | lhr | anaiza777@gmail.com | 897 |
+-----+-----+-----+-----+-----+
3 rows in set (0.02 sec)

mysql> select *from manufacturer;
+-----+-----+-----+-----+
| manufacturer_id | Name | Address | Contact_info |
+-----+-----+-----+-----+
| 101 | Ali | skp | 0321-1234567 |
| 102 | Alyan | lhr | 0321-1234569 |
| 103 | Arman | fsd | 0321-1234568 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

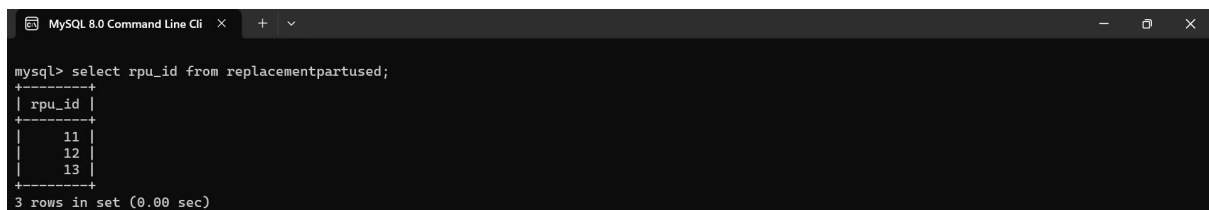
mysql> select *from product;
+-----+-----+-----+-----+-----+
| product_id | Model_no | Product_type | Price | Serial_no |
+-----+-----+-----+-----+-----+
| 301 | MOD-A1 | Smartphone | 899.99 | SN-XYZ001 |
| 302 | MOD-B2 | Laptop | 1499.50 | SN-ABC202 |
| 303 | MOD-C3 | Headphones | 199.00 | SN-QWE345 |
+-----+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

SELECT one/multiple column

This command is beneficial at that support while you are dealing with to come out the single one and the multiple attributes from a table rather than all of the attributes. Only a certain or specific value will be come out by this.

Syntax;

SELECT column_name FROM table_name;



```
mysql> select rpu_id from replacementpartused;
+-----+
| rpu_id |
+-----+
| 11 |
| 12 |
| 13 |
+-----+
3 rows in set (0.00 sec)
```

ALTER command

Used to alter the table and including other attributes or inserts more data into the table.

Syntax:

ALTER TABLE table_name ADD column_name CONSTRAINT;

- Alter the table attributes by using the command;
ALTER TABLE servicerequest ADD c_id int(6);

```
mysql> CREATE TABLE servicerequest(service_req_id INT PRIMARY KEY,issue_desc VARCHAR(100),Request_date DATE);
Query OK, 0 rows affected (0.05 sec)

mysql> ALTER TABLE servicerequest ADD product_id INT(6), c_id INT(6);
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 'c_id INT(6)' at line 1
mysql> ALTER TABLE servicerequest ADD product_id INT(6);
Query OK, 0 rows affected, 1 warning (0.08 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> ALTER TABLE servicerequest ADD c_id int(6);
Query OK, 0 rows affected, 1 warning (0.07 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> DESC servicerequest;
```

Field	Type	Null	Key	Default	Extra
service_req_id	int	NO	PRI	NULL	
issue_desc	varchar(100)	YES		NULL	
Request_date	date	YES		NULL	
product_id	int	YES		NULL	
c_id	int	YES		NULL	

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

Suppose we want to include the already present table EMPLOYEE the PRIMARY KEY. Then for the sack of that purpose we use:

AS COMMAND on MySQL

This is basically used to make an alias of the attribute. It's helpful where you want to access the already attribute name with the new one as you want.

Syntax:

SELECT column_name AS ALIAS FROM table_name;

```
mysql> select coverage_details AS information from warrenty;
```

information
Parts and labor for 1 year
Full coverage for 3 years
Accidental damage + 24/7 support

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

DISTINCT COMMAND

This command is used for duplicity removal from your table if exist.

Syntax:

SELECT DISTINCT column_name FROM table_name;

```
mysql> select DISTINCT Expertise from technician;
```

Expertise
Aerospace Avionics Repair
Vintage Watch Restoration
Oil Rig Industrial Sensors

```
3 rows in set (0.02 sec)
```

WHERE clause in SQL

Where command is work as such like from the condition .

Syntax:

```
SELECT *FROM table_name WHERE column='VALUES';
```



```
mysql> select *from customer WHERE c_id='2';
```

c_id	Name	Address	Email	PostalCode
2	zayan	skp	zayan08@gmail.com	567

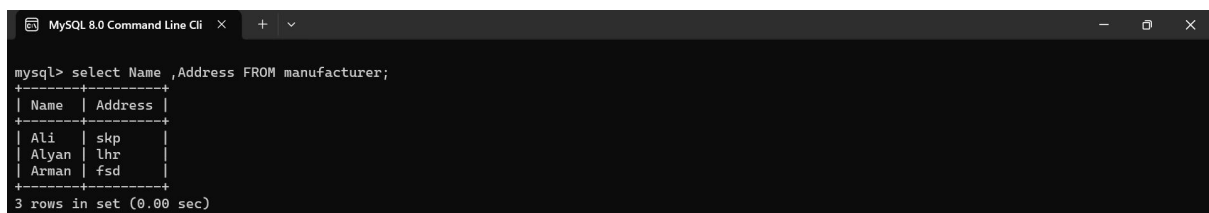
```
1 row in set (0.02 sec)
```

ORDER BY command

It's used for the ascending or descending order sorting. By default ascending sort is done by the compiler.

Syntax:

```
SELECT column_name FROM table_name ORDER BY column_2 desc;
```



```
mysql> select Name ,Address FROM manufacturer;
```

Name	Address
Ali	skp
Alyan	lhr
Arman	fsd

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

Operators in SQL

Arithmetic operators including (+,-,*,/).

Syntax:

```
SELECT column_name OPERATOR FROM table_name WHERE column_name='VALUES';
```

Or

```
SELECT column_name FROM table_name OPERATOR any_operation;
```

```
MySQL 8.0 Command Line Cli x + v
mysql> select cost*12 FROM repairhistory;
+-----+
| cost*12 |
+-----+
| 1079.88 |
| 546.00  |
| 3588.00 |
+-----+
3 rows in set (0.02 sec)

mysql> select cost+100 AS Expenditure FROM repairhistory;
+-----+
| Expenditure |
+-----+
| 189.99      |
| 145.50      |
| 399.00      |
+-----+
3 rows in set (0.00 sec)

mysql> select cost-50 FROM repairhistory;
+-----+
| cost-50 |
+-----+
| 39.99   |
| -4.50   |
| 249.00  |
+-----+
3 rows in set (0.02 sec)
```

Or

```
MySQL 8.0 Command Line Cli x + v
mysql> select cost+100 FROM repairhistory WHERE repair_history_id='1002';
+-----+
| cost+100 |
+-----+
| 145.50   |
+-----+
1 row in set (0.00 sec)

mysql> select cost+100*12 AS Total_Cost FROM repairhistory;
+-----+
| Total_Cost |
+-----+
| 1289.99    |
| 1245.50    |
| 1499.00    |
+-----+
3 rows in set (0.00 sec)
```

Relational operators

These operators including (>,<,>=,<=,==,!=).

Syntax:

```
SELECT column_name OPERATOR FROM table_name WHERE
column_name='VALUES';
```

Or

```
SELECT column_name FROM table_name WHERE column
OPERATOR;
```

```

MySQL 8.0 Command Line Cli x + v
mysql> select cost FROM repairhistory WHERE cost>45;
+-----+
| cost |
+-----+
| 89.99 |
| 45.50 |
| 299.00 |
+-----+
3 rows in set (0.02 sec)

mysql> select cost FROM repairhistory WHERE cost<89;
+-----+
| cost |
+-----+
| 45.50 |
+-----+
1 row in set (0.00 sec)

```

Logical Operators

AND ,OR ,NOT.

SQL QUERIES BY AND,OR OPERATOR

```

MySQL 8.0 Command Line Cli x + v
mysql> select *from product WHERE(product_id=301 AND Model_no='MOD-A1');
+-----+-----+-----+-----+-----+
| product_id | Model_no | Product_type | Price | Serial_no |
+-----+-----+-----+-----+-----+
| 301 | MOD-A1 | Smartphone | 899.99 | SN-XYZ001 |
+-----+-----+-----+-----+-----+
1 row in set (0.02 sec)

mysql> select price from product WHERE (product_id=302 OR Price>=899);
+-----+
| price |
+-----+
| 899.99 |
| 1499.50 |
+-----+
2 rows in set (0.00 sec)

mysql> select warrenty_type FROM warrenty WHERE(warrenty_id!=1);
+-----+
| warrenty_type |
+-----+
| Extended Warranty |
| Premium Protection |
+-----+
2 rows in set (0.00 sec)

```

USE OF MULTIPLE AND OPERATOR

```

MySQL 8.0 Command Line Cli x + v
mysql> select *from manufacturer WHERE (manufacturer_id='101' && Address='skp');
+-----+-----+-----+-----+
| manufacturer_id | Name | Address | Contact_info |
+-----+-----+-----+-----+
| 101 | Ali | skp | 0321-1234567 |
+-----+-----+-----+-----+
1 row in set, 1 warning (0.00 sec)

mysql> select *from customer WHERE (c_id='2' OR Address='skp');
+-----+-----+-----+-----+-----+
| c_id | Name | Address | Email | PostalCode |
+-----+-----+-----+-----+-----+
| 2 | zayan | skp | zayan08@gmail.com | 567 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select Model_no FROM product WHERE (product_id='303' AND Price>='199' AND Price<='1499' AND Price BETWEEN 199 AND 1499);
+-----+
| Model_no |
+-----+
| MOD-C3 |
+-----+
1 row in set (0.02 sec)

mysql> select repair_history_id FROM repairhistory WHERE (repair_history_id='1003' OR cost BETWEEN 89 AND 299);
+-----+
| repair_history_id |
+-----+
| 1001 |
| 1003 |
+-----+
2 rows in set (0.00 sec)

```

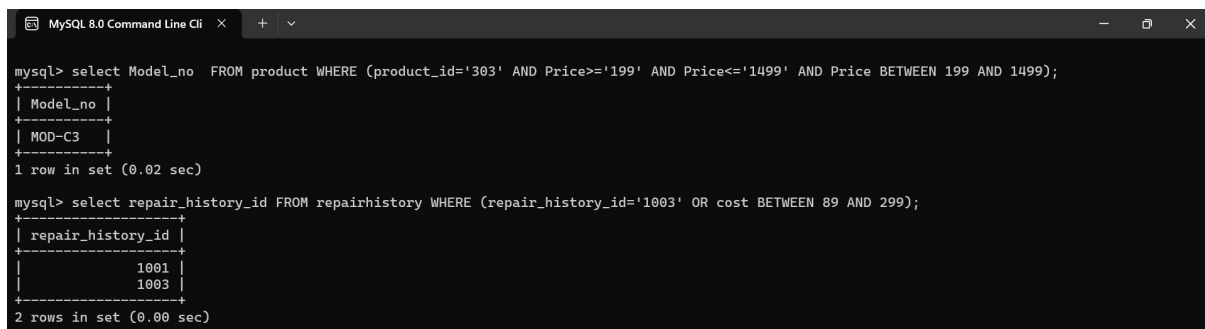
AND BETWEEN

It's used where you want to evaluate the values from the range. Within that specific range. SQL COMMANDS.

Syntax:

```
SELECT *FROM table_name WHERE(condition_1 AND  
condition_2 BETWEEN condition_3);
```

As you can use any column_name rather than the asterisk inside the query.



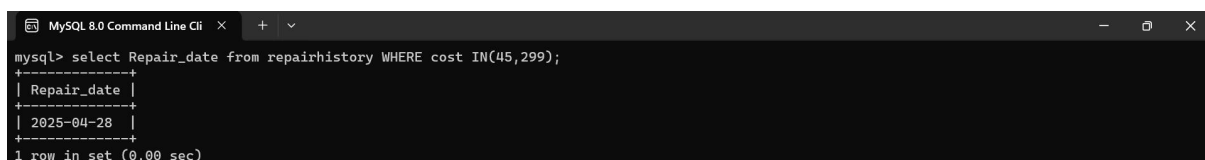
```
mysql> select Model_no FROM product WHERE (product_id='303' AND Price>='199' AND Price<='1499' AND Price BETWEEN 199 AND 1499);  
+-----+  
| Model_no |  
+-----+  
| MOD-C3   |  
+-----+  
1 row in set (0.02 sec)  
  
mysql> select repair_history_id FROM repairhistory WHERE (repair_history_id='1003' OR cost BETWEEN 89 AND 299);  
+-----+  
| repair_history_id |  
+-----+  
| 1001              |  
| 1003              |  
+-----+  
2 rows in set (0.00 sec)
```

IN COMMAND

It's uses to come out the certain value from the list of the table. It's basically used to reduce length which is taken by multiple AND or operators.

Syntax:

```
SELECT column_name FROM table_name WHERE column_name  
IN(val_1 ,val_2);
```



```
mysql> select Repair_date from repairhistory WHERE cost IN(45,299);  
+-----+  
| Repair_date |  
+-----+  
| 2025-04-28  |  
+-----+  
1 row in set (0.00 sec)
```

LIKE COMMAND

As if you want to evaluate a specific personality from a table whose spelling you are specifying in the query or any other name whose end or start or at any middle element you'll be mentioned by you. This command is beneficial at that spot while you want to take a name whose character you'll not be known is advanced.

Wildcard Characters:

- %(Represents the single ,null or multiple characters)

- `_` (Represents a single character)

Syntax:

`SELECT column_1, column_2 FROM table_name WHERE column_name LIKE pattern;`

```
MySQL 8.0 Command Line Cli x + v
mysql> select Name FROM customer WHERE Name LIKE 'A%';
+-----+
| Name |
+-----+
| Azka  |
| Anaiza|
+-----+
2 rows in set (0.00 sec)

mysql> select Name FROM customer WHERE Name LIKE '_A%';
+-----+
| Name |
+-----+
| zayan |
+-----+
1 row in set (0.00 sec)
```

NULL

```
MySQL 8.0 Command Line Cli x + v
mysql> select Name from customer WHERE PostalCode IS NULL;
Empty set (0.00 sec)
```

As there is no column contains NULL VALUES so it returns 'empty set', but command is correct.

AGGREGATION FUNCTIONS:

- `COUNT(COLUMN_NAME)`
- `COUNT(*)`
- `AVG(COLUMN_NAME)`
- `MAX(COLUMN_NAME)`
- `MIN(COLUMN_NAME)`
- `SUM(COUMN_NAME)`

COUNT COMMAND

```
MySQL 8.0 Command Line Cli x + v
mysql> select COUNT(*) FROM customer;
+-----+
| COUNT(*) |
+-----+
| 3        |
+-----+
1 row in set (0.00 sec)

mysql> select COUNT(Price) FROM product;
+-----+
| COUNT(Price) |
+-----+
| 3            |
+-----+
1 row in set (0.02 sec)
```

- `COUNT(*)` will display the all the columns which is include in your tables including the null values and also the duplicate values.

- **COUNT(column_name)** will display ONLY the singly values and not null values.

AVERAGE FUNCTIONS

Syntax:

SELECT AGGREGATE_FUNCTION(column_name) FROM table_name;

AVERAGE

```
mysql> select AVG(cost) FROM repairhistory;
+-----+
| AVG(cost) |
+-----+
| 144.830000 |
+-----+
1 row in set (0.00 sec)
```

MIN , MAX , SUM

```
mysql> select MAX(cost) FROM repairhistory;
+-----+
| MAX(cost) |
+-----+
| 299.00 |
+-----+
1 row in set (0.02 sec)

mysql> select MIN(cost) FROM repairhistory;
+-----+
| MIN(cost) |
+-----+
| 45.50 |
+-----+
1 row in set (0.02 sec)

mysql> select SUM(Price) FROM product;
+-----+
| SUM(Price) |
+-----+
| 2598.49 |
+-----+
1 row in set (0.00 sec)

mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date ,Description;
+-----+
| MAX(cost) |
+-----+
| 89.99 |
| 45.50 |
| 299.00 |
+-----+
3 rows in set (0.00 sec)
```

GROUP BY

```
mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date ,Description;
+-----+
| MAX(cost) |
+-----+
| 89.99 |
| 45.50 |
| 299.00 |
+-----+
3 rows in set (0.00 sec)
```

HAVING COMMAND

It's uses as such to apply the conditions indirectly which is not done with the help of where command as while you are dealing with the GROUP BY in sql.

So for the sake of that purpose you have to use the HAVING to filter out of data or attributes as you want from the table.

For example:

```
mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date HAVING COUNT(repair_history_id>2);
+-----+
| MAX(cost) |
+-----+
| 89.99     |
| 45.50     |
| 299.00    |
+-----+
3 rows in set (0.02 sec)

mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date HAVING COUNT(repair_history_id)>3;
Empty set (0.00 sec)

mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date HAVING COUNT(repair_history_id)>2;
Empty set (0.00 sec)

mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date HAVING COUNT(repair_history_id)>1;
Empty set (0.00 sec)

mysql> select MAX(cost) FROM repairhistory GROUP BY Repair_date HAVING COUNT(repair_history_id)>1;
+-----+
| MAX(cost) |
+-----+
| 89.99     |
| 45.50     |
| 299.00    |
+-----+
3 rows in set (0.00 sec)
```

Join

Types of join are;

- Inner join
- Right join
- Left join
- Outer join

select*from customer RIGHT JOIN product on product.ProductName=customer. Name;

```
mysql> select*from customer RIGHT JOIN product on product.ProductName=customer.Name;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| C_id | Name | Address | Email | PostalCode | Product_id | Model_Number | Product_Type | Price | SerialNumber | ProductName |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| NULL | NULL | NULL | NULL | NULL | 101 | 2025 | Basic | 180000.00 | LAP123 | Laptop |
| NULL | NULL | NULL | NULL | NULL | 205 | 2023 | Extended | 80000.00 | PHM456 | Iphone X |
| NULL | NULL | NULL | NULL | NULL | 511 | 2024 | Limited | 50000.00 | WH329 | Apple Watch |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

select*from customer LEFT JOIN product on product.ProductName=customer. Name;

```
mysql> select*from customer LEFT JOIN product on product.ProductName=customer.Name;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| C_id | Name | Address | Email | PostalCode | Product_id | Model_Number | Product_Type | Price | SerialNumber | ProductName |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Ahmad | Skp | ahmad01@gmail.com | 987 | NULL | NULL | NULL | NULL | NULL | NULL |
| 2 | Hassan | Lhr | hassan02@gmail.com | 6887 | NULL | NULL | NULL | NULL | NULL | NULL |
| 3 | Zameen | Hfz | zameen03@gmail.com | 5678 | NULL | NULL | NULL | NULL | NULL | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

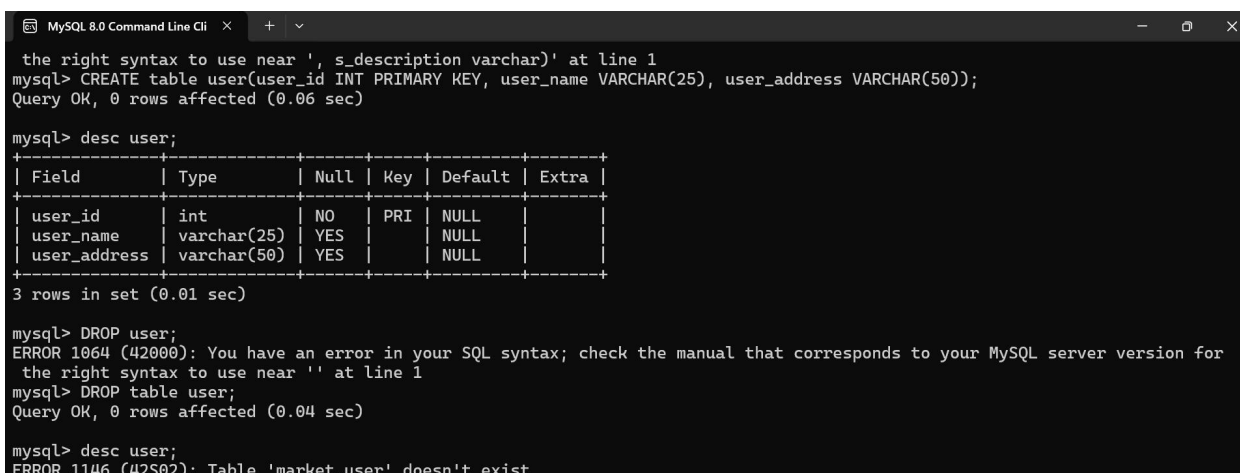
Some commands for inner join which are not actually applicable on this database are;

- `select product.ProductName,repairhistory.Cost from product INNER JOIN repairhistory on repairhistory.Cost=product.ProductName;`
- `select product.ProductName,repairhistory.Cost from product INNER JOIN repairhistory on repairhistory.Cost>product.ProductName;`
- `select product.ProductName,repairhistory.Cost from product INNER JOIN repairhistory on repairhistory.Cost<product.ProductName;`

Delete Table/Database

- **Table:** Suppose we have created the table mistakenly we can delete it using following command.

Command : `DROP TABLE user;`



```
MySQL 8.0 Command Line Cli x + v
the right syntax to use near ', s_description varchar)' at line 1
mysql> CREATE table user(user_id INT PRIMARY KEY, user_name VARCHAR(25), user_address VARCHAR(50));
Query OK, 0 rows affected (0.06 sec)

mysql> desc user;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| user_id    | int       | NO   | PRI | NULL    |       |
| user_name  | varchar(25) | YES  |     | NULL    |       |
| user_address | varchar(50) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)

mysql> DROP user;
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 '' at line 1
mysql> DROP table user;
Query OK, 0 rows affected (0.04 sec)

mysql> desc user;
ERROR 1146 (42S02): Table 'market.user' doesn't exist
```

- **Database:** Suppose we have created the database mistakenly we can delete it using following command.

Command: `DROP DATABASE market;`

```
mysql> SHOW databases;
+-----+
| Database |
+-----+
| dept     |
| information_schema |
| market   |
| mysql    |
| performance_schema |
| sys      |
+-----+
6 rows in set (0.00 sec)

mysql> DROP database market;
Query OK, 9 rows affected (0.19 sec)

mysql> describe tables;
ERROR 1046 (3D000): No database selected
mysql>
```
