

SQL assignment_1 - Nucleusteq consulting pvt. ltd

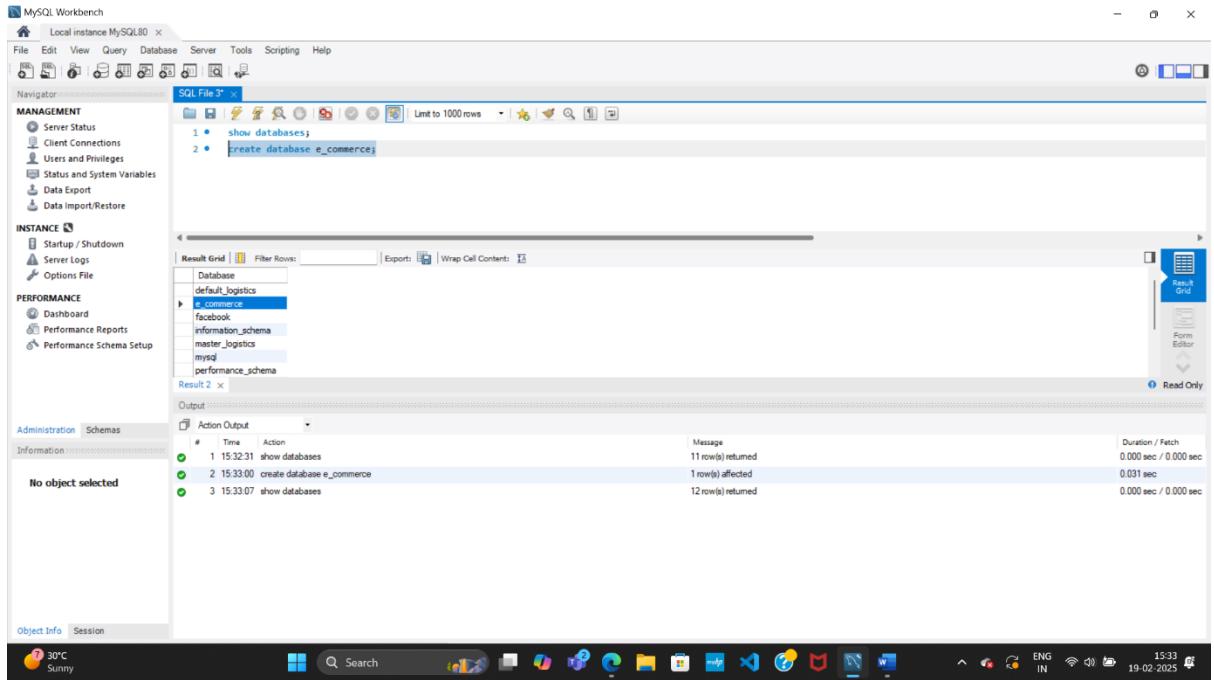
Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

1. Create Database e_commerce

SQL query:- create database db_name;



The screenshot shows the MySQL Workbench interface. In the SQL Editor tab, the following SQL command is run:

```
1 • show databases;
2 • create database e_commerce;
```

In the Result Grid, the output shows the creation of the 'e_commerce' database and lists other existing databases:

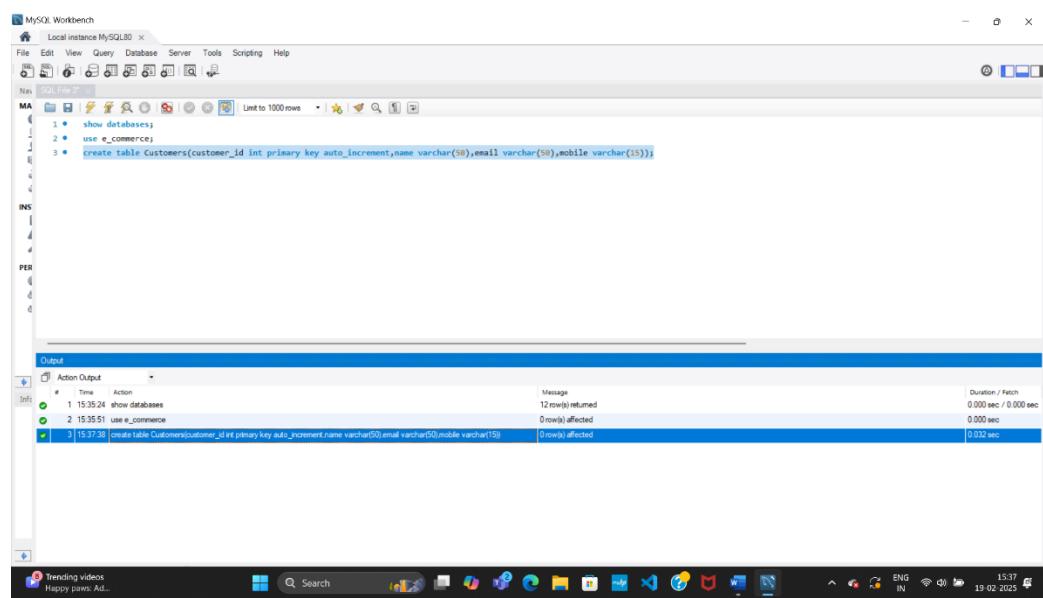
Database
default_logistics
e_commerce
facebook
information_schema
master_logistics
mysql
performance_schema

The Action Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	15:32:31	show databases	11 row(s) returned	0.000 sec / 0.000 sec
2	15:33:00	create database e_commerce	1 row(s) affected	0.031 sec
3	15:33:07	show databases	12 row(s) returned	0.000 sec / 0.000 sec

2. Create Customer Tables:

SQL query:- create table Customers(customer_id int primary key auto_increment,name varchar(50),email varchar(50),mobile varchar(15));



The screenshot shows the MySQL Workbench interface. In the SQL Editor tab, the following SQL command is run:

```
1 • show databases;
2 • use e_commerce;
3 • create table Customers(customer_id int primary key auto_increment,name varchar(50),email varchar(50),mobile varchar(15));
```

The Action Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	15:35:24	show databases	12 row(s) returned	0.000 sec / 0.000 sec
2	15:35:51	use e_commerce	0 row(s) affected	0.000 sec
3	15:37:30	create table Customers(customer_id int primary key auto_increment,name varchar(50),email varchar(50),mobile varchar(15))	0 row(s) affected	0.032 sec

SQL assignment_1 - Nucleusteq consulting pvt. ltd

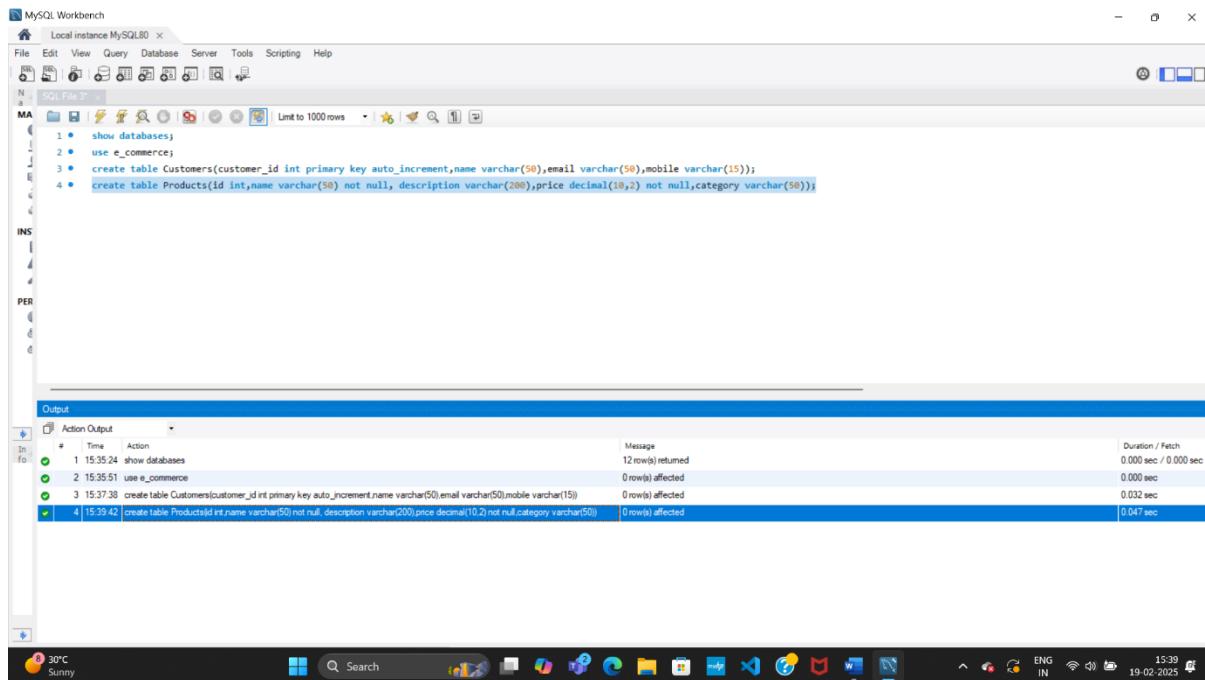
Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

3. Create Products Tables:

SQL query:- create table Products(id int,name varchar(50) not null,description varchar(200),price decimal(10,2) not null,category varchar(50));



The screenshot shows the MySQL Workbench interface with the following details:

- MA (Main Area):** Contains the SQL code:

```
1 • show databases;
2 • use e_commerce;
3 • create table Customers(customer_id int primary key auto_increment, name varchar(50), email varchar(50), mobile varchar(15));
4 • create table Products(id int, name varchar(50) not null, description varchar(200), price decimal(10,2) not null, category varchar(50));
```
- INS (Information Schema):** Displays the structure of the created tables.
- PER (Performance Schema):** Displays the performance metrics for the queries.
- Output:** Shows the execution log with the following entries:

#	Time	Action	Message	Duration / Fetch
1	15:35:24	show databases	12 row(s) returned	0.000 sec / 0.000 sec
2	15:35:51	use e_commerce	0 row(s) affected	0.000 sec
3	15:37:38	create table Customers(customer_id int primary key auto_increment, name varchar(50), email varchar(50), mobile varchar(15))	0 row(s) affected	0.032 sec
4	15:39:42	create table Products(id int, name varchar(50) not null, description varchar(200), price decimal(10,2) not null, category varchar(50))	0 row(s) affected	0.047 sec

4. Modify Tables(using Alter keyword):

1. Add not null on name and email in the Customers table:-

SQL query:-

```
alter table Customers modify column name varchar(50)  
not null;
```

```
alter table customers modify column email varchar(50)  
not null;
```

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the following SQL code:

```
-- 3. Modify Tables(using Alter keyword):--  
-- 1. a. Add not null on name and email in the Customers table  
10 • alter table Customers modify column name varchar(50) not null;  
11 • alter table customers modify column email varchar(50) not null;  
12 • desc Customers;
```
- Result Grid:** Shows the structure of the Customers table with columns: customer_id (int, NO, PRI, auto_increment), name (varchar(50), NO), email (varchar(50), NO), and mobile (varchar(15), YES).
- Action Output:** Displays the execution log with rows numbered 1 to 8, showing actions like 'show databases', 'use e_commerce', and 'alter table' commands, along with their messages, duration, and fetch times.

2. Add unique key on email in the Customers table:-

SQL query:- alter table customers modify column email varchar(50) unique key;

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the following SQL code:

```
-- b. Add unique key on email in the Customers table--  
15 • alter table customers modify column email varchar(50) unique key;  
16 • desc customers;
```
- Result Grid:** Shows the structure of the Customers table with columns: customer_id (int, NO, PRI, auto_increment), name (varchar(50), NO), email (varchar(50), YES, UNQ), and mobile (varchar(15), YES).
- Action Output:** Displays the execution log with rows numbered 1 to 10, showing actions like 'show databases', 'use e_commerce', and 'alter table' commands, along with their messages, duration, and fetch times.

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

3. Add column age in the Customers table

SQL query:- alter table customers add column age int;

The screenshot shows the MySQL Workbench interface. In the SQL editor, the following SQL code is written:

```
-- c. Add column age in the Customers table--

The results pane shows the structure of the 'customers' table, which now includes a new column 'age' of type int with a default value of null.



The output pane displays the history of actions taken:



| #  | Time     | Action                                                                                                                             | Message                                                     | Duration / Fetch      |
|----|----------|------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|-----------------------|
| 3  | 15:37:38 | create table Customers(customer_id int primary key auto_increment,name varchar(50),email varchar(50),mobile varchar(15))           | 0 row(s) affected                                           | 0.032 sec             |
| 4  | 15:39:42 | create table Products(id int,name varchar(50) not null,description varchar(200),price decimal(10,2) not null,category varchar(50)) | 0 row(s) affected                                           | 0.047 sec             |
| 5  | 15:47:56 | alter table Customer modify column name varchar(50) not null                                                                       | Error Code: 1146: Table 'e_commerce.customer' doesn't exist | 0.000 sec             |
| 6  | 15:48:16 | alter table Customers modify column name varchar(50) not null                                                                      | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.094 sec             |
| 7  | 15:48:17 | alter table customers modify column email varchar(50) not null                                                                     | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.063 sec             |
| 8  | 15:48:17 | desc Customers                                                                                                                     | 4 row(s) returned                                           | 0.015 sec / 0.000 sec |
| 9  | 15:50:20 | alter table customers modify column email varchar(50) unique key                                                                   | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.093 sec             |
| 10 | 15:50:20 | desc customers                                                                                                                     | 4 row(s) returned                                           | 0.000 sec / 0.000 sec |
| 11 | 15:52:29 | alter table customers add column age int                                                                                           | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.031 sec             |
| 12 | 15:52:29 | desc customers                                                                                                                     | 5 row(s) returned                                           | 0.000 sec / 0.000 sec |


```

4. Change column name from id to product_id in the Products table;

SQL query:- alter table Products rename column id to product_id;

The screenshot shows the MySQL Workbench interface. In the SQL editor, the following SQL code is written:

```
-- d. Change column name from id to product_id in the Products table--

The results pane shows the structure of the 'Products' table, which now has a column named 'product_id'.



The output pane displays the history of actions taken:



| #  | Time     | Action                                                           | Message                                                     | Duration / Fetch      |
|----|----------|------------------------------------------------------------------|-------------------------------------------------------------|-----------------------|
| 5  | 15:47:56 | alter table Customer modify column name varchar(50) not null     | Error Code: 1146: Table 'e_commerce.customer' doesn't exist | 0.000 sec             |
| 6  | 15:48:16 | alter table Customers modify column name varchar(50) not null    | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.094 sec             |
| 7  | 15:48:17 | alter table customers modify column email varchar(50) not null   | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.063 sec             |
| 8  | 15:48:17 | desc Customers                                                   | 4 row(s) returned                                           | 0.015 sec / 0.000 sec |
| 9  | 15:50:20 | alter table customers modify column email varchar(50) unique key | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.093 sec             |
| 10 | 15:50:20 | desc customers                                                   | 4 row(s) returned                                           | 0.000 sec / 0.000 sec |
| 11 | 15:52:29 | alter table customers add column age int                         | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.031 sec             |
| 12 | 15:52:29 | desc customers                                                   | 5 row(s) returned                                           | 0.000 sec / 0.000 sec |
| 13 | 15:54:34 | alter table Products rename column id to product_id              | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0      | 0.047 sec             |
| 14 | 15:54:34 | desc Products                                                    | 5 row(s) returned                                           | 0.000 sec / 0.000 sec |


```

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

5. Add primary key and auto increment on product_id in the Products table.

SQL query:- alter table products modify column product_id int primary key auto_increment;

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for connection, queries, results, etc.
- Query Editor:** Shows the following SQL code:

```
-- e. Add primary key and auto increment on product_id in the Products table--  
alter table products modify column product_id int primary key auto_increment;  
desc products;
```
- Result Grid:** Displays the structure of the 'products' table:

Field	Type	Null	Key	Default	Extra
product_id	int	NO	PRI	0	auto_increment
name	varchar(50)	NO			
description	varchar(200)	YES			
price	decimal(10,2)	NO			
category	varchar(50)	YES			
- Action Output:** Shows the history of actions taken:

#	Time	Action	Message	Duration / Fetch
7	15:48:17	alter table customers modify column email varchar(50) not null	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.063 sec
8	15:48:17	desc Customers	4 row(s) returned	0.015 sec / 0.000 sec
9	15:50:20	alter table customers modify column email varchar(50) unique key	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.093 sec
10	15:50:20	desc customers	4 row(s) returned	0.000 sec / 0.000 sec
11	15:52:29	alter table customers add column age int	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
12	15:52:29	desc customers	5 row(s) returned	0.000 sec / 0.000 sec
13	15:54:34	alter table Products rename column id to product_id	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
14	15:54:34	desc Products	5 row(s) returned	0.000 sec / 0.000 sec
15	15:56:01	alter table products modify column product_id int primary key auto_increment	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.094 sec
16	15:56:01	desc products	5 row(s) returned	0.000 sec / 0.000 sec
- System Status:** Shows 30°C, Sunny, 15:56, ENG IN, 19-02-2025.

6. Change datatype of description from varchar to text in the Products table

SQL query:- alter table products modify column description text(200)

The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Toolbar:** Standard MySQL icons for connection, queries, results, etc.
- Query Editor:** Shows the following SQL code:

```
-- f. Change datatype of description from varchar to text in the Products table  
alter table products modify column description text(200);  
desc products;
```
- Result Grid:** Displays the structure of the 'products' table after the modification:

Field	Type	Null	Key	Default	Extra
product_id	int	NO	PRI	0	auto_increment
name	varchar(50)	NO			
description	text(200)	YES			
price	decimal(10,2)	NO			
category	varchar(50)	YES			
- Action Output:** Shows the history of actions taken:

#	Time	Action	Message	Duration / Fetch
9	15:59:20	alter table customers modify column email varchar(50) unique key	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.079 sec
10	15:59:20	desc customers	4 row(s) returned	0.000 sec / 0.000 sec
11	15:59:20	alter table customers add column age int	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.033 sec
12	15:59:29	desc customers	5 row(s) returned	0.000 sec / 0.000 sec
13	15:54:34	alter table Products rename column id to product_id	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
14	15:56:01	desc products	5 row(s) returned	0.000 sec / 0.000 sec
15	15:56:01	alter table products modify column product_id int primary key auto_increment	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.094 sec
16	15:56:01	desc products	5 row(s) returned	0.000 sec / 0.000 sec
17	15:59:21	desc products	5 row(s) returned	0.000 sec / 0.000 sec
- System Status:** Shows 30°C, Sunny, 15:59, ENG IN, 19-02-2025.

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

5. Create table Order:

i can't create a table named order as its a reserved word in sql that why i used a backticks so that sql server treat "order" as an identifier.

SQL query:- create table `order`(`order_id` int primary key auto_increment, customer_id int ,product_id int ,quantity int not null, order_date date not null, status enum('Pending','Success','Cancel'), payment_method enum('Credit','Debit"UPI"),total_amount decimal(10,2) not null, foreign key(customer_id) references Customers(customer_id));

The screenshot shows the MySQL Workbench interface with the following details:

- SQL Editor:** Contains the SQL code for creating the 'order' table. The code uses backticks around 'order' to avoid a reserved keyword error. It includes columns for order_id (auto-increment), customer_id, product_id, quantity (not null), order_date (date type), status (enum type with values 'Pending', 'Success', 'Cancel'), payment_method (enum type with values 'Credit', 'Debit"UPI'), and total_amount (decimal type with precision 10, scale 2). A foreign key constraint is added for customer_id referencing the customer_id in the Customers table.
- Result Grid:** Shows the structure of the 'order' table with columns: Field, Type, Null, Key, Default, Extra. The columns listed are order_id, customer_id, product_id, quantity, order_date, status, payment_method, and total_amount.
- Action Output:** Displays the execution log with 20 entries, showing the time, action, message, and duration for each step of the table creation process.
- System Bar:** At the bottom, it shows the Windows taskbar with various application icons and system status indicators like battery level and network connection.

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

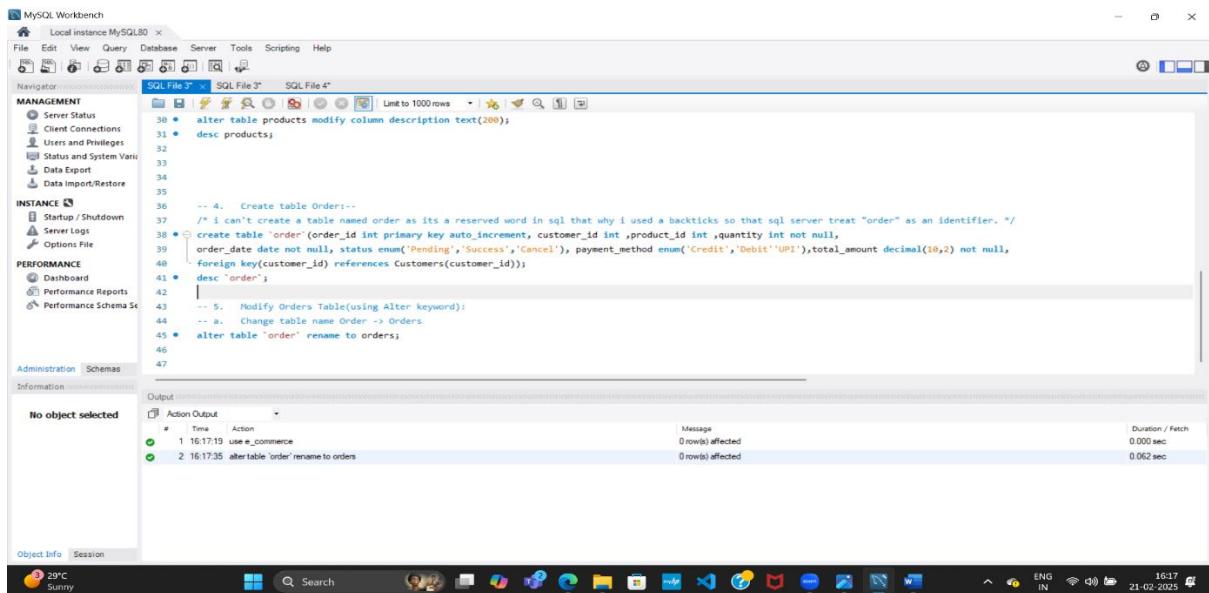
Chetanpal.work@gmail.com

Date:-21/02/2025

6. Modify Orders Table(using Alter keyword):

a. Change table name Order -> Orders

SQL query:- alter table `order` rename to orders;



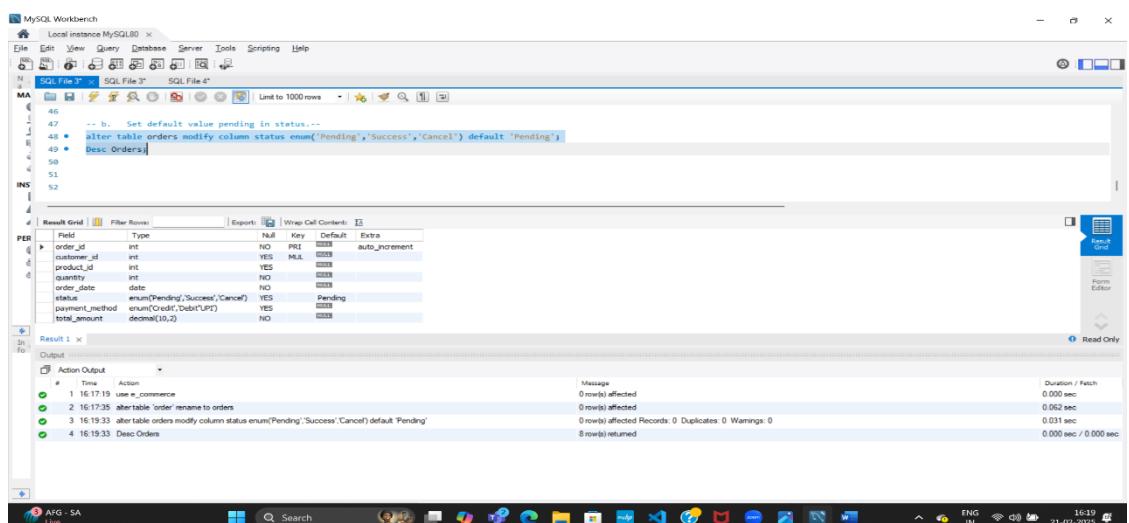
```
30 • alter table products modify column description text(200);
31 • desc products;
32
33
34
35
36 -- 4. Create table Order:--
37 /* I can't create a table named order as its a reserved word in sql that why i used a backticks so that sql server treat "order" as an identifier. */
38 • create table `order` (order_id int primary key auto_increment, customer_id int ,product_id int ,quantity int not null,
39 order_date date not null, status enum('Pending','Success','Cancel'), payment_method enum('Credit','Debit','UPI'),total_amount decimal(10,2) not null,
40 foreign key(customer_id) references Customers(customer_id));
41 • desc `order`;
42
43 -- 5. Modify Orders Table(using Alter keyword):
44 -- a. Change table name Order -> Orders
45 • alter table `order` rename to orders;
46
47
```

Action Output

#	Time	Action	Message	Duration / Fetch
1	16:17:13	use e_commerce	0 row(s) affected	0.000 sec
2	16:17:35	alterable `order` rename to orders	0 row(s) affected	0.062 sec

b. Set default value pending in status

SQL query:- alter table orders modify column status enum('Pending','Success','Cancel') default 'Pending';



```
46
47 -- b. Set default value pending in status:-
48 • alter table orders modify column status enum('Pending','Success','Cancel') default 'Pending';
49 • Desc Orders;
50
51
52
```

Result 1

Field	Type	Null	Key	Default	Extra
order_id	int	NO	PRI	auto_increment	
customer_id	int	YES	MUL		
product_id	int	YES			
quantity	int	NO			
order_date	date	NO			
status	enum('Pending','Success','Cancel')	YES		Pending	
payment_method	enum('Credit','Debit','UPI')	YES			
total_amount	decimal(10,2)	NO			

Action Output

#	Time	Action	Message	Duration / Fetch
1	16:17:19	use e_commerce	0 row(s) affected	0.000 sec
2	16:17:35	alterable `order` rename to orders	0 row(s) affected	0.062 sec
3	16:19:33	alterable orders modify column status enum('Pending','Success','Cancel') default 'Pending'	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
4	16:19:33	Desc Orders	8 rows(s) returned	0.000 sec / 0.000 sec

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

- c. Modify payment_method ENUM to add one more value: 'COD'

SQL query:- alter table orders modify column

payment_method enum('Credit','Debit','UPI','COD');

The screenshot shows the MySQL Workbench interface with the following details:

- Queries:**

```
-- b. Set default value pending in status...
alter table orders modify column status enum('Pending','Success','Cancel') default 'Pending';
Desc Orders;
```

```
-- c. Modify payment_method ENUM to add one more value: 'COD'--
alter table orders modify column payment_method enum('Credit','Debit','UPI','COD');
desc orders;
```
- Result Grid:** Shows the structure of the 'orders' table with columns: order_id, customer_id, product_id, quantity, order_date, status, and payment_method. The payment_method column is defined as enum('Credit','Debit','UPI','COD').
- Action Output:** Displays the execution log with 6 entries, each showing the time, action, message, and duration.
- Environment:** The taskbar at the bottom shows various application icons, and the system tray indicates the date as 21-02-2025.

- d. Make product id as foreign key

SQL query:- alter table orders add foreign key(product_id) references products(product_id);

The screenshot shows the MySQL Workbench interface with the following details:

- Queries:**

```
-- d. Make product id as foreign key--
alter table orders add foreign key(product_id) references Products(product_id);
desc orders;
```
- Result Grid:** Shows the structure of the 'orders' table with columns: order_id, customer_id, product_id, quantity, order_date, status, payment_method, and total_amount. The product_id column is now defined as a foreign key referencing the product_id column in the Products table.
- Action Output:** Displays the execution log with 9 entries, each showing the time, action, message, and duration.
- Environment:** The taskbar at the bottom shows various application icons, and the system tray indicates the date as 21-02-2025.

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

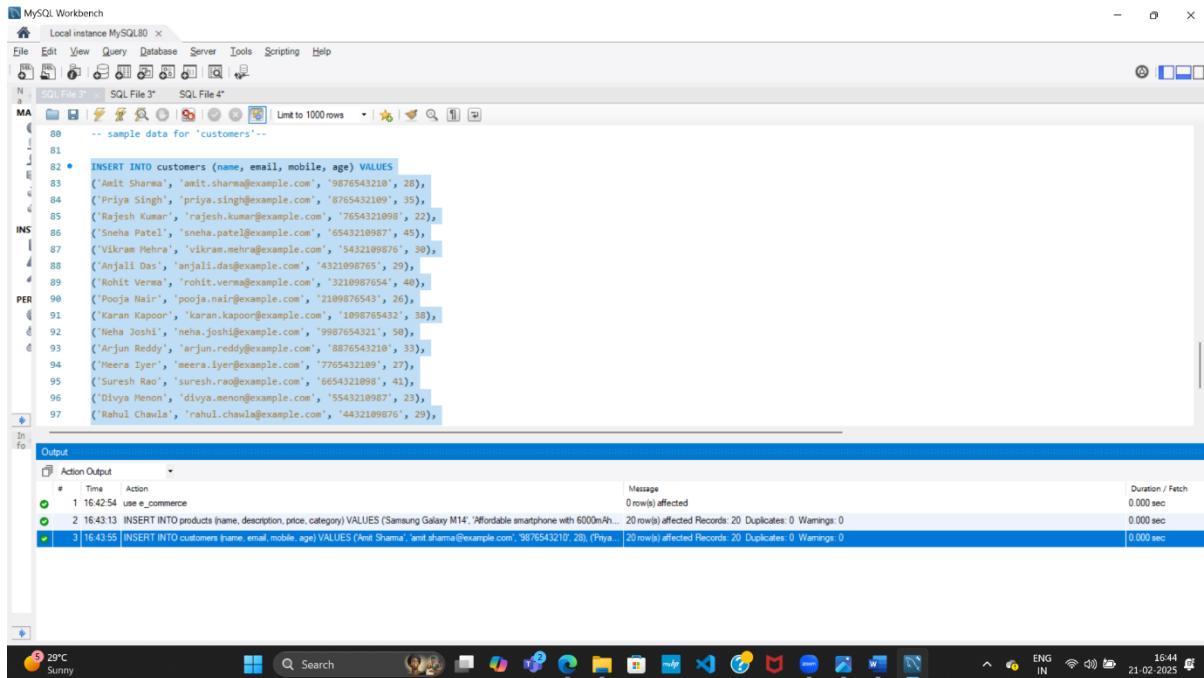
Date:-21/02/2025

7. Insert 20 sample records in all the tables.

a. 20 sample data for 'customers'

SQL query:- INSERT INTO customers (name, email, mobile, age) VALUES

('Amit Sharma', 'amit.sharma@example.com', '9876543210', 28);



The screenshot shows the MySQL Workbench interface. In the SQL editor pane, there is a multi-line SQL script. The first line is a comment: '-- sample data for 'customers'--'. The second line is the start of an INSERT INTO statement. The third line contains the values for the first row: ('Amit Sharma', 'amit.sharma@example.com', '9876543210', 28). Subsequent lines show more rows being inserted, each with a unique name, email, mobile number, and age. The output pane at the bottom shows the results of the execution, including the number of rows affected (20) and the duration (0.000 sec).

```
-- sample data for 'customers'--  
INSERT INTO customers (name, email, mobile, age) VALUES  
('Amit Sharma', 'amit.sharma@example.com', '9876543210', 28),  
('Priya Singh', 'priya.singh@example.com', '8765432109', 35),  
('Rajesh Kumar', 'rajesh.kumar@example.com', '7654321098', 22),  
('Sneha Patel', 'sneha.patel@example.com', '6543210987', 45),  
('Vikram Mehta', 'vikram.mehta@example.com', '5432109876', 30),  
('Anjali Das', 'anjali.das@example.com', '4321098765', 29),  
('Rohit Verma', 'rohit.verma@example.com', '3210987654', 40),  
('Pooja Nair', 'pooja.nair@example.com', '2109876543', 26),  
('Karan Kapoor', 'karan.kapoor@example.com', '1098765432', 38),  
('Neha Joshi', 'neha.joshi@example.com', '9987654321', 50),  
('Arjun Reddy', 'arjun.reddy@example.com', '8876543210', 33),  
('Meera Iyer', 'meera.iyer@example.com', '7765432109', 27),  
('Suresh Rao', 'suresh.rao@example.com', '6654321098', 41),  
('Divya Menon', 'divya.menon@example.com', '5543210987', 23),  
('Rahul Chawla', 'rahul.chawla@example.com', '4432109876', 29),
```

Action	Time	Action	Message	Duration / Fetch
1	16:42:54	use_e_commerce	0 row(s) affected	0.000 sec
2	16:43:13	INSERT INTO products (name, description, price, category) VALUES ('Samsung Galaxy M14', 'Affordable smartphone with 6000mAh battery', 13999.00, 'Electronics')	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
3	16:43:55	INSERT INTO customers (name, email, mobile, age) VALUES ('Amit Sharma', 'amit.sharma@example.com', '9876543210', 28)	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec

b. 20 sample data for products table;

SQL query:- INSERT INTO products (name, description, price, category) VALUES

('Samsung Galaxy M14', 'Affordable smartphone with 6000mAh battery', 13999.00, 'Electronics');

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

The screenshot shows the MySQL Workbench interface with the following details:

- File**: Local instance MySQL80 X
- Database**: SQL File 3*, SQL File 4*
- Table**: products
- SQL Query**:

```
55
56 -- d. Make product_id as Foreign key--  
57 ALTER TABLE products ADD CONSTRAINT fk_product_id FOREIGN KEY (product_id) REFERENCES categories(category_id);  
58 INSERT INTO products (name, description, price, category) VALUES  
59 ('Samsung Galaxy M14', 'Affordable smartphone with 6500mAH battery', 13999.00, 'Electronics'),  
60 ('Realme Narzo 60', 'Mid-range smartphone with AMOLED display', 17999.00, 'Electronics'),  
61 ('Infinix Hot 11', 'Wireless Bluetooth earphones with deep bass', 999.00, 'Electronics'),  
62 ('Xiaomi Pad 5', 'High-performance tablet with stylus support', 27999.00, 'Electronics'),  
63 ('Canon EOS 200D II', 'Compact DSLR with 18-55mm lens and flip screen', 26999.00, 'Electronics'),  
64 ('LG Refrigerator', 'Double-door refrigerator with smart inverter', 26999.00, 'Home Appliances'),  
65 ('TIFB Microwave Oven', 'Convection microwave with auto-cook menu', 8990.00, 'Home Appliances'),  
66 ('Philips Air Fryer', '750W air fryer with added basket', 3499.00, 'Home Appliances'),  
67 ('Philips Safe Box', '3-drawer safe box with keypad', 2299.00, 'Home Appliances'),  
68 ('Godrej Dining Table', '6-seater dining set with glass top', 14999.00, 'Furniture'),  
69 ('IKEA Chair', 'Ergonomic office chair with lumbar support', 4999.00, 'Furniture'),  
70 ('Urban Ladder Bookshelf', '5-tier bookshelf with modern design', 7999.00, 'Furniture'),  
71 ('Myntra T-shirt', 'Cotton blend t-shirt for men', 1399.00, 'Clothing'),  
72 ('Levi's Jean', 'Skin-fit stretch jeans for men', 3299.00, 'Clothing'),  
73 ('Woodland Jacket', 'Water-resistant jacket for outdoor use', 5999.00, 'Clothing')
```
- Output**:

Action	Output	Message	Duration / Fetch
1 14:43:54	Use e_commerce	0 rows(s) affected.	0.000 sec
2 14:43:13	INSERT INTO products (name, description, price, category) VALUES ('Samsung Galaxy M14', 'Affordable smartphone with 6500mAH battery', 13999.00, 'Electronics')	20 record(s) affected, 20 Duplicate(s), 0 Warning(s)	0.000 sec

c. sample data for 'orders'

```
SQL query:- INSERT INTO orders (customer_id, product_id, quantity, order_date, status, payment_method, total_amount) VALUES (1, 1, 1, '2024-01-10', 'Success', 'Credit', 13999.00);
```

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

SQL File 3* SQL File 4*

MA Unit to 1000 rows ▾

INS

PER

DEL

In

fo

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	16:42:54	use e_commerce	0 row(s) affected	0.000 sec
2	16:43:13	INSERT INTO products (name, description, price, category) VALUES ('Samsung Galaxy M14', 'Affordable smartphone with 6000mAh battery and 90Hz display', 1299.00, 'Smartphones')	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
3	16:43:55	INSERT INTO customers (name, email, mobile, age) VALUES ('Amrit Sharma', 'amrit.sharma@example.com', '9876543210', 28)	(Replay) 20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec
4	16:46:09	INSERT INTO orders (customer_id, product_id, quantity, order_date, status, payment_method, total_amount) VALUES (1, 1, 1, '2024-01-10', 'Success', 'Credit', 1299.00)	Error Code: 1265. Data truncated for column 'status' at row 12	0.015 sec
5	16:47:45	INSERT INTO orders (customer_id, product_id, quantity, order_date, status, payment_method, total_amount) VALUES (1, 1, 1, '2024-01-10', 'Success', 'Credit', 1299.00)	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0	0.000 sec

29°C Sunny

Search

ENG IN

16:48 21-02-2025

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

8. Performing following queries:

- Count the number of products as product_count in each category.

SQL query:-
SELECT category, COUNT(*) AS product_count
FROM products GROUP BY category;

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the SQL query:

```
-- a. Count the number of products as product_count in each category.
SELECT category, COUNT(*) AS product_count FROM products GROUP BY category;
```
- Result Grid:** Displays the output of the query:

category	product_count
Electronics	5
Books & References	4
Furniture	4
Clothing	4
Accessories	4
- Action Output:** Shows the execution log with various SQL statements and their results, including insertions into the e_commerce database.

- Retrieve all products that belong to the 'Electronics' category, have a price between \$50(4250inr) and \$500(42500inr), and whose name contains the letter 'a'.

SQL query:-
SELECT * FROM products WHERE category = 'Electronics' AND price BETWEEN 4250 AND 42500 AND name LIKE '%a%';

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the SQL query:

```
-- b. Retrieve all products that belong to the 'Electronics' category, have a price between $50(4250inr) and $500(42500inr), and whose name contains the letter 'a'.
SELECT * FROM products WHERE category = 'Electronics' AND price BETWEEN 4250 AND 42500 AND name LIKE '%a%';
```
- Result Grid:** Displays the output of the query:

product_id	name	description	price	category
1	Samsung Galaxy M14	Affordable smartphone with 6000mAh battery	13999.00	Electronics
2	Redmi Note 60	High-range smartphone with AMOLED display	27999.00	Electronics
3	AS Pad 2	High-performance tablet with stylus support	10999.00	Electronics
- Action Output:** Shows the execution log with various SQL statements and their results, including insertions into the e_commerce database.

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

- c. Get the top 5 most expensive products in the 'Electronics' category, skipping the first 2.

SQL query:-
SELECT * FROM products WHERE category = 'Electronics' ORDER BY price DESC LIMIT 2, 5;

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the SQL query: "SELECT * FROM products WHERE category = 'Electronics' ORDER BY price DESC LIMIT 2, 5;"
- Results Grid:** Displays the results of the query, showing 5 rows of product information:

product_id	name	description	price	category
2	Realme Narzo 60	Mid-range smartphone with AMOLED display	17999.00	Electronics
1	Samsung Galaxy M14	Affordable smartphone with 6000mAh battery	19999.00	Electronics
3	Boat Rockerz 255	Wireless Bluetooth headphones with deep bass	2999.00	Electronics
4	Logitech G502		2999.00	Electronics

- d. Retrieve customers who have not placed any orders.

SQL query:-
SELECT * FROM customers WHERE
customer_id NOT IN (SELECT DISTINCT customer_id
FROM orders);

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Contains the SQL query: "SELECT * FROM customers WHERE customer_id NOT IN (SELECT DISTINCT customer_id FROM orders);"
- Results Grid:** Displays the results of the query, showing 0 rows of customer information.

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

- e. Find the average total amount spent by each customer.

SQL query:-
SELECT c.customer_id, c.name,
AVG(o.total_amount) AS avg_spent FROM customers c
JOIN orders o ON c.customer_id = o.customer_id GROUP BY
BY c.customer_id;

```
-- e. Average total amount spent by each customer--  
SELECT c.customer_id, c.name, AVG(o.total_amount) AS avg_spent FROM customers c JOIN orders o ON c.customer_id = o.customer_id GROUP BY c.customer_id;
```

Customer_Id	Name	Avg_Spent
1	Ankit Sharma	13999.000000
2	Priyanshu Singh	39999.000000
3	Rajesh Kumar	9999.000000
4	Sneha Patel	27999.000000
5	Vikash Verma	49999.000000
6	Arjali Das	26999.000000
7	Rohit Verma	17980.000000
8	Pooja Nair	34999.000000
9	Kiran Kapoor	28999.000000

- f. Get the products that have a price less than the average price of all products.

SQL query:-
SELECT * FROM products WHERE price < (SELECT
AVG(price) FROM products);

```
-- f. Products with price less than average price of all products  
SELECT * FROM products WHERE price < (SELECT AVG(price) FROM products);
```

product_id	name	description	price	category
3	Boat Rockerz 255	Wireless Bluetooth earphones with deep bass	999.00	Electronics
6	IPL Microwave Oven	Cooking microwave with auto-cook menu	8990.00	Kitchen Appliances
7	Hipster Hand Grinder	750W hand grinder with adjustable settings	1299.00	Kitchen Appliances
11	DEFA Chair	Ergonomic office chair with lumbar support	4999.00	Furniture
12	Urban Ladder Bookshelf	5-tier bookshelf with modern design	7999.00	Furniture
13	Pubg T-shirt	Gaming t-shirt for men	1299.00	Clothing
14	Levi's Jeans	Slim-fit stretch jeans	3299.00	Clothing
15	Woodland Jacket	Water-resistant jacket for outdoor use	9999.00	Clothing
16	Addidas Sneakers	Running shoes with cloudfoam cushioning	4999.00	Clothing

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

- g. Calculate the total quantity of products ordered by each customer:

SQL query:-
SELECT c.customer_id, c.name,
SUM(o.quantity) AS total_quantity FROM customers c JOIN
orders o ON c.customer_id = o.customer_id GROUP BY
c.customer_id;

The screenshot shows the MySQL Workbench interface with several queries run in the SQL editor. The last query executed is:

```
SELECT c.customer_id, c.name, SUM(o.quantity) AS total_quantity FROM customers c JOIN orders o ON c.customer_id = o.customer_id GROUP BY c.customer_id;
```

The results are displayed in a grid:

customer_id	name	total_quantity
1	Ankit Sharma	1
2	Priyanshu	2
3	Rajesh Kumar	1
4	Sneha Patel	1
5	Vikram Mehta	1
6	Arijit Deka	1
7	Rohit Verma	2
8	Poops Nair	1
9	Karan Kapoor	1

The output pane shows the execution history with timestamps and message details.

- h. List all orders along with customer name and product name.

SQL query:-
SELECT o.order_id, c.name AS customer_name, p.name AS product_name, o.order_date
FROM orders o JOIN customers c ON o.customer_id =
c.customer_id JOIN products p ON o.product_id =
p.product_id;

SQL assignment_1 - Nucleusteq consulting pvt. ltd

Chetan pal

Chetanpal.work@gmail.com

Date:-21/02/2025

-- E. Total quantity of products ordered by each customer
153 --
154 SELECT c.customer_id, c.name, sum(o.quantity) AS total_quantity FROM customers c JOIN orders o ON c.customer_id = o.customer_id GROUP BY c.customer_id;
155 --
156 -- h. List all orders with customer name and product name
157 --
158 SELECT o.order_id, c.name AS customer_name, p.name AS product_name, o.order_date FROM orders o JOIN customers c ON o.customer_id = c.customer_id JOIN products p ON o.product_id = p.product_id;
159 --
160 --
161 --
162 --
163 --
164 --
165 --
166 --
167 --
168 --
169 --
170 --
171 --
172 --
173 --
174 --
175 --
176 --
177 --
178 --
179 --
180 --
181 --
182 --
183 --
184 --
185 --
186 --
187 --
188 --
189 --
190 --
191 --
192 --
193 --
194 --
195 --
196 --
197 --
198 --
199 --
200 --
201 --
202 --
203 --
204 --
205 --
206 --
207 --
208 --
209 --
210 --
211 --
212 --
213 --
214 --
215 --
216 --
217 --
218 --
219 --
220 --
221 --
222 --
223 --
224 --
225 --
226 --
227 --
228 --
229 --
230 --
231 --
232 --
233 --
234 --
235 --
236 --
237 --
238 --
239 --
240 --
241 --
242 --
243 --
244 --
245 --
246 --
247 --
248 --
249 --
250 --
251 --
252 --
253 --
254 --
255 --
256 --
257 --
258 --
259 --
260 --
261 --
262 --
263 --
264 --
265 --
266 --
267 --
268 --
269 --
270 --
271 --
272 --
273 --
274 --
275 --
276 --
277 --
278 --
279 --
280 --
281 --
282 --
283 --
284 --
285 --
286 --
287 --
288 --
289 --
290 --
291 --
292 --
293 --
294 --
295 --
296 --
297 --
298 --
299 --
300 --
301 --
302 --
303 --
304 --
305 --
306 --
307 --
308 --
309 --
310 --
311 --
312 --
313 --
314 --
315 --
316 --
317 --
318 --
319 --
320 --
321 --
322 --
323 --
324 --
325 --
326 --
327 --
328 --
329 --
330 --
331 --
332 --
333 --
334 --
335 --
336 --
337 --
338 --
339 --
340 --
341 --
342 --
343 --
344 --
345 --
346 --
347 --
348 --
349 --
350 --
351 --
352 --
353 --
354 --
355 --
356 --
357 --
358 --
359 --
360 --
361 --
362 --
363 --
364 --
365 --
366 --
367 --
368 --
369 --
370 --
371 --
372 --
373 --
374 --
375 --
376 --
377 --
378 --
379 --
380 --
381 --
382 --
383 --
384 --
385 --
386 --
387 --
388 --
389 --
390 --
391 --
392 --
393 --
394 --
395 --
396 --
397 --
398 --
399 --
400 --
401 --
402 --
403 --
404 --
405 --
406 --
407 --
408 --
409 --
410 --
411 --
412 --
413 --
414 --
415 --
416 --
417 --
418 --
419 --
420 --
421 --
422 --
423 --
424 --
425 --
426 --
427 --
428 --
429 --
430 --
431 --
432 --
433 --
434 --
435 --
436 --
437 --
438 --
439 --
440 --
441 --
442 --
443 --
444 --
445 --
446 --
447 --
448 --
449 --
450 --
451 --
452 --
453 --
454 --
455 --
456 --
457 --
458 --
459 --
460 --
461 --
462 --
463 --
464 --
465 --
466 --
467 --
468 --
469 --
470 --
471 --
472 --
473 --
474 --
475 --
476 --
477 --
478 --
479 --
480 --
481 --
482 --
483 --
484 --
485 --
486 --
487 --
488 --
489 --
490 --
491 --
492 --
493 --
494 --
495 --
496 --
497 --
498 --
499 --
500 --
501 --
502 --
503 --
504 --
505 --
506 --
507 --
508 --
509 --
510 --
511 --
512 --
513 --
514 --
515 --
516 --
517 --
518 --
519 --
520 --
521 --
522 --
523 --
524 --
525 --
526 --
527 --
528 --
529 --
530 --
531 --
532 --
533 --
534 --
535 --
536 --
537 --
538 --
539 --
540 --
541 --
542 --
543 --
544 --
545 --
546 --
547 --
548 --
549 --
550 --
551 --
552 --
553 --
554 --
555 --
556 --
557 --
558 --
559 --
560 --
561 --
562 --
563 --
564 --
565 --
566 --
567 --
568 --
569 --
570 --
571 --
572 --
573 --
574 --
575 --
576 --
577 --
578 --
579 --
580 --
581 --
582 --
583 --
584 --
585 --
586 --
587 --
588 --
589 --
590 --
591 --
592 --
593 --
594 --
595 --
596 --
597 --
598 --
599 --
600 --
601 --
602 --
603 --
604 --
605 --
606 --
607 --
608 --
609 --
610 --
611 --
612 --
613 --
614 --
615 --
616 --
617 --
618 --
619 --
620 --
621 --
622 --
623 --
624 --
625 --
626 --
627 --
628 --
629 --
630 --
631 --
632 --
633 --
634 --
635 --
636 --
637 --
638 --
639 --
640 --
641 --
642 --
643 --
644 --
645 --
646 --
647 --
648 --
649 --
650 --
651 --
652 --
653 --
654 --
655 --
656 --
657 --
658 --
659 --
660 --
661 --
662 --
663 --
664 --
665 --
666 --
667 --
668 --
669 --
670 --
671 --
672 --
673 --
674 --
675 --
676 --
677 --
678 --
679 --
680 --
681 --
682 --
683 --
684 --
685 --
686 --
687 --
688 --
689 --
690 --
691 --
692 --
693 --
694 --
695 --
696 --
697 --
698 --
699 --
700 --
701 --
702 --
703 --
704 --
705 --
706 --
707 --
708 --
709 --
710 --
711 --
712 --
713 --
714 --
715 --
716 --
717 --
718 --
719 --
720 --
721 --
722 --
723 --
724 --
725 --
726 --
727 --
728 --
729 --
730 --
731 --
732 --
733 --
734 --
735 --
736 --
737 --
738 --
739 --
740 --
741 --
742 --
743 --
744 --
745 --
746 --
747 --
748 --
749 --
750 --
751 --
752 --
753 --
754 --
755 --
756 --
757 --
758 --
759 --
760 --
761 --
762 --
763 --
764 --
765 --
766 --
767 --
768 --
769 --
770 --
771 --
772 --
773 --
774 --
775 --
776 --
777 --
778 --
779 --
780 --
781 --
782 --
783 --
784 --
785 --
786 --
787 --
788 --
789 --
790 --
791 --
792 --
793 --
794 --
795 --
796 --
797 --
798 --
799 --
800 --
801 --
802 --
803 --
804 --
805 --
806 --
807 --
808 --
809 --
810 --
811 --
812 --
813 --
814 --
815 --
816 --
817 --
818 --
819 --
820 --
821 --
822 --
823 --
824 --
825 --
826 --
827 --
828 --
829 --
830 --
831 --
832 --
833 --
834 --
835 --
836 --
837 --
838 --
839 --
840 --
841 --
842 --
843 --
844 --
845 --
846 --
847 --
848 --
849 --
850 --
851 --
852 --
853 --
854 --
855 --
856 --
857 --
858 --
859 --
860 --
861 --
862 --
863 --
864 --
865 --
866 --
867 --
868 --
869 --
870 --
871 --
872 --
873 --
874 --
875 --
876 --
877 --
878 --
879 --
880 --
881 --
882 --
883 --
884 --
885 --
886 --
887 --
888 --
889 --
890 --
891 --
892 --
893 --
894 --
895 --
896 --
897 --
898 --
899 --
900 --
901 --
902 --
903 --
904 --
905 --
906 --
907 --
908 --
909 --
910 --
911 --
912 --
913 --
914 --
915 --
916 --
917 --
918 --
919 --
920 --
921 --
922 --
923 --
924 --
925 --
926 --
927 --
928 --
929 --
930 --
931 --
932 --
933 --
934 --
935 --
936 --
937 --
938 --
939 --
940 --
941 --
942 --
943 --
944 --
945 --
946 --
947 --
948 --
949 --
950 --
951 --
952 --
953 --
954 --
955 --
956 --
957 --
958 --
959 --
960 --
961 --
962 --
963 --
964 --
965 --
966 --
967 --
968 --
969 --
970 --
971 --
972 --
973 --
974 --
975 --
976 --
977 --
978 --
979 --
980 --
981 --
982 --
983 --
984 --
985 --
986 --
987 --
988 --
989 --
990 --
991 --
992 --
993 --
994 --
995 --
996 --
997 --
998 --
999 --
1000 --

- i. Find products that have never been ordered.

SQL query:- SELECT * FROM products WHERE product_id

NOT IN (SELECT DISTINCT product_id FROM orders);

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

SQL File 5* SQL File 3* SQL File 4*

MA Limit to 1000 rows

155
156 -- h. List all orders with customer name and product name
157 • SELECT o.order_id, c.name AS customer_name, p.name AS product_name, o.order_date FROM orders o JOIN customers c ON o.customer_id = c.customer_id JOIN products p ON o.product_id = p.product_id;
158
159 -- i. Products never ordered
160 • SELECT * FROM products WHERE product_id NOT IN (SELECT DISTINCT product_id FROM orders);
INS 161
162
163

Result Grid Filter Rows! Edit: Export/Import: Wrap Cell Content:

product_id	name	description	price	category
NULL	NULL	NULL	NULL	NULL

PER

In Info

products 10 x

Output

Action Output	#	Time	Action	Message	Duration / Fetch
9 16:53:14	SELECT * FROM products WHERE category = 'Electronics' AND price BETWEEN 4250 AND 42500 AND name LIKE "%a%" LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec		
10 16:54:11	SELECT * FROM products WHERE category = 'Electronics' ORDER BY price DESC LIMIT 2, 5	3 row(s) returned	0.000 sec / 0.000 sec		
11 16:55:02	SELECT * FROM customers WHERE customer_id NOT IN (SELECT DISTINCT customer_id FROM orders) LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec		
12 16:55:40	SELECT c.customer_id, c.name, AVG(o.total_amount) AS avg_spent FROM customers c JOIN orders o ON c.customer_id = o.customer_id GROUP BY c.customer_id	20 row(s) returned	0.000 sec / 0.000 sec		
13 16:56:26	SELECT * FROM products WHERE price < (SELECT AVG(price) FROM products) LIMIT 0, 1000	13 row(s) returned	0.000 sec / 0.000 sec		
14 16:57:10	SELECT c.customer_id, c.name, SUM(o.quantity) AS total_quantity FROM customers c JOIN orders o ON c.customer_id = o.customer_id GROUP BY c.customer_id	20 row(s) returned	0.000 sec / 0.000 sec		
15 16:57:35	SELECT o.order_id, c.name AS customer_name, p.name AS product_name, o.order_date FROM orders o JOIN customers c ON o.customer_id = c.customer_id JOIN products p ON o.product_id = p.product_id WHERE o.order_id NOT IN (SELECT DISTINCT order_id FROM orders) LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec		
16 16:58:00	SELECT * FROM products WHERE product_id NOT IN (SELECT DISTINCT product_id FROM orders) LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec		

5 AFG - SA Live Search

ENG IN 16:58 21-02-2025