Assignment

Assignment of Basic Database Concepts

Basic Database Concepts Assignment - 1

Name:	Ali Ahasan
Position:	Intern
Tracking Number:	J2DFEEB
Team:	InnovX Team of Business Automation Limited
Session:	Basic Database Concepts with MySQL

Assignment Overview:

In this assignment, you will design, create, and manage a simple e-commerce database. You will perform various tasks, including creating tables, inserting data, and writing SQL queries to retrieve and manipulate the data. This assignment will help you understand the practical aspects of using MySQL in an e-commerce context.

Task:

- Design a database schema for a simple e-commerce application, including tables for products, customers, orders with minimum column.
- Insert sample data into the e-commerce database created.
- Prepare a query to Retrieve customer information based on a specific order.
- Prepare a query to find the total number of orders placed by each customer.
- Prepare a query to get the list of top 10 customers who have ordered the most times.
- List of products that have been ordered the most number of times with the number of orders.

Instructions:

1. Complete all the query as outlined above and attach query & output screenshot in single file.
2. Name the file as "name_yourId_db_assignment".
3. Submit the file through the provided Google Form.

Table of Contents

E-commerce database

Schema design

Database Schema

Tables:

1. **Products**: This table holds all the information about the items available for purchase.
   - ❖ product_id (INT, Primary Key)
   - ❖ product_name (VARCHAR)
   - ❖ price (DECIMAL)
2. **Customers**: This Table Stores which are all the details about the customers in specific.
   - ❖ customer_id (INT, Primary Key)
   - ❖ customer_name (VARCHAR)
   - ❖ email (VARCHAR)
3. **Orders**: For this table the details that consists of the orders which are placed by the customers.
   - ❖ order_id (INT, Primary Key)
   - ❖ customer_id (INT, Foreign Key from Customers)
   - ❖ order_date (DATE)
4. **Order_Items**: This one table represents what kind of items that are in each of their order.
   - ❖ order_item_id (INT, Primary Key)
   - ❖ order_id (INT, Foreign Key from Orders)
   - ❖ product_id (INT, Foreign Key from Products)
   - ❖ quantity (INT)

Diagram:

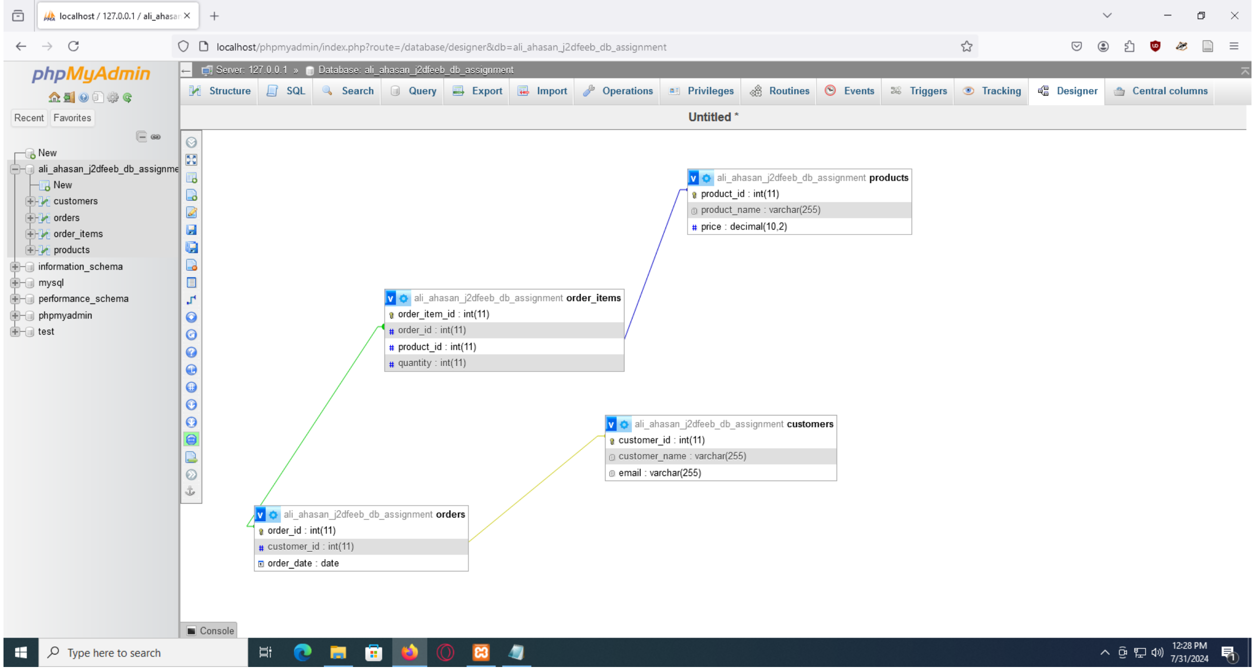


Table Making:

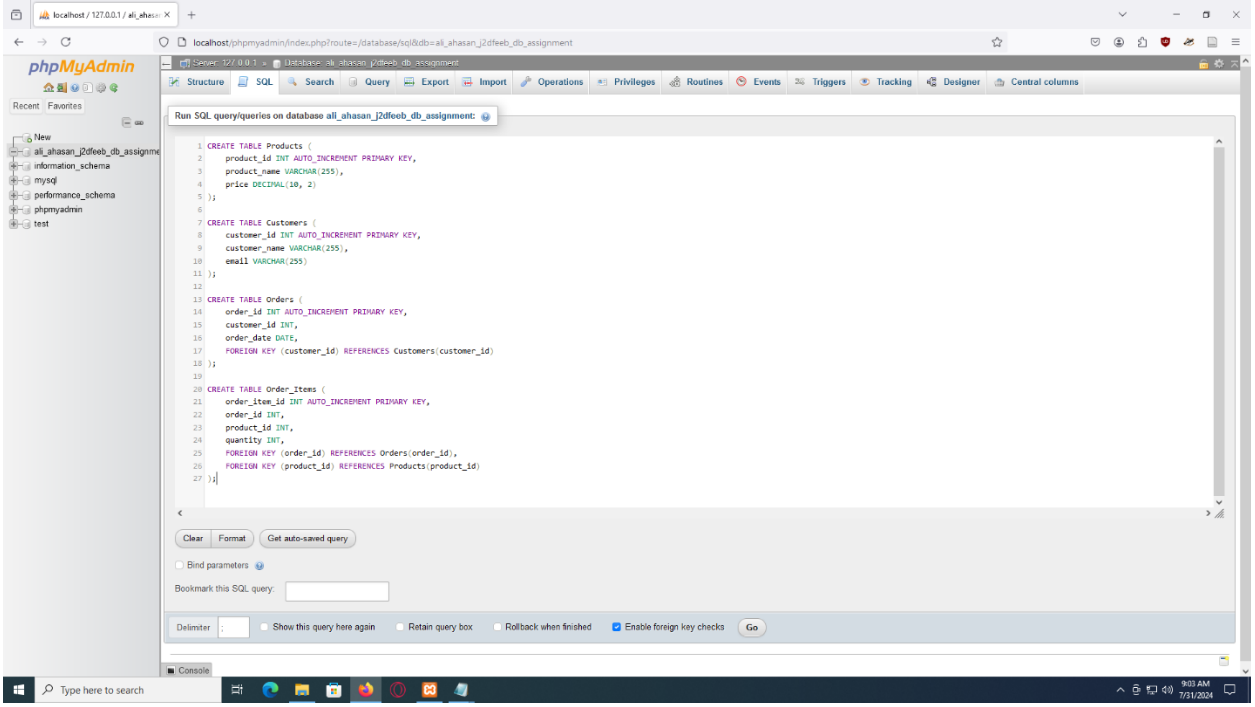
Using DDL (Data Definition Language) to make the table.

Code:

```
CREATE TABLE Products (
    product_id INT AUTO_INCREMENT PRIMARY KEY,
    product_name VARCHAR(255),
    price DECIMAL(10, 2)
);

CREATE TABLE Customers (
    customer_id INT AUTO_INCREMENT PRIMARY KEY,
    customer_name VARCHAR(255),
    email VARCHAR(255)
);

CREATE TABLE Orders (
    order_id INT AUTO_INCREMENT PRIMARY KEY,
    customer_id INT,
    order_date DATE,
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id)
);

CREATE TABLE Order_Items (
    order_item_id INT AUTO_INCREMENT PRIMARY KEY,
    order_id INT,
    product_id INT,
    quantity INT,
    FOREIGN KEY (order_id) REFERENCES Orders(order_id),
    FOREIGN KEY (product_id) REFERENCES Products(product_id)
);
```

Input:

1



Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:

```sql
CREATE TABLE Products (
    product_id INT AUTO_INCREMENT PRIMARY KEY,
    product_name VARCHAR(255),
    price DECIMAL(10, 2)
);

CREATE TABLE Customers (
    customer_id INT AUTO_INCREMENT PRIMARY KEY,
    customer_name VARCHAR(255),
    email VARCHAR(255)
);

CREATE TABLE Orders (
    order_id INT AUTO_INCREMENT PRIMARY KEY,
    customer_id INT,
    order_date DATE,
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id)
);

CREATE TABLE Order_Items (
    order_item_id INT AUTO_INCREMENT PRIMARY KEY,
    order_id INT,
    product_id INT,
    quantity INT,
    FOREIGN KEY (order_id) REFERENCES Orders(order_id),
    FOREIGN KEY (product_id) REFERENCES Products(product_id)
);
```

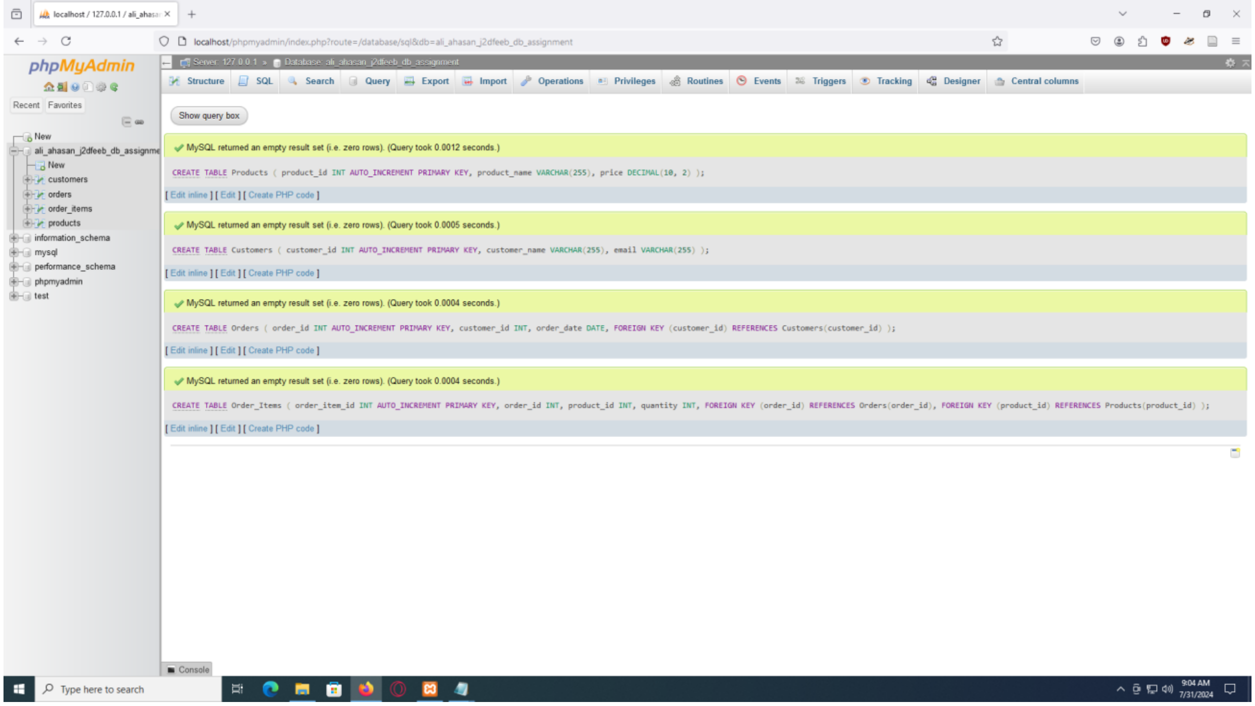
□ Bind parameters

Bookmark this SQL query:

Delimiter ; □ Show this query here again □ Retain query box □ Rollback when finished ☑ Enable foreign key checks

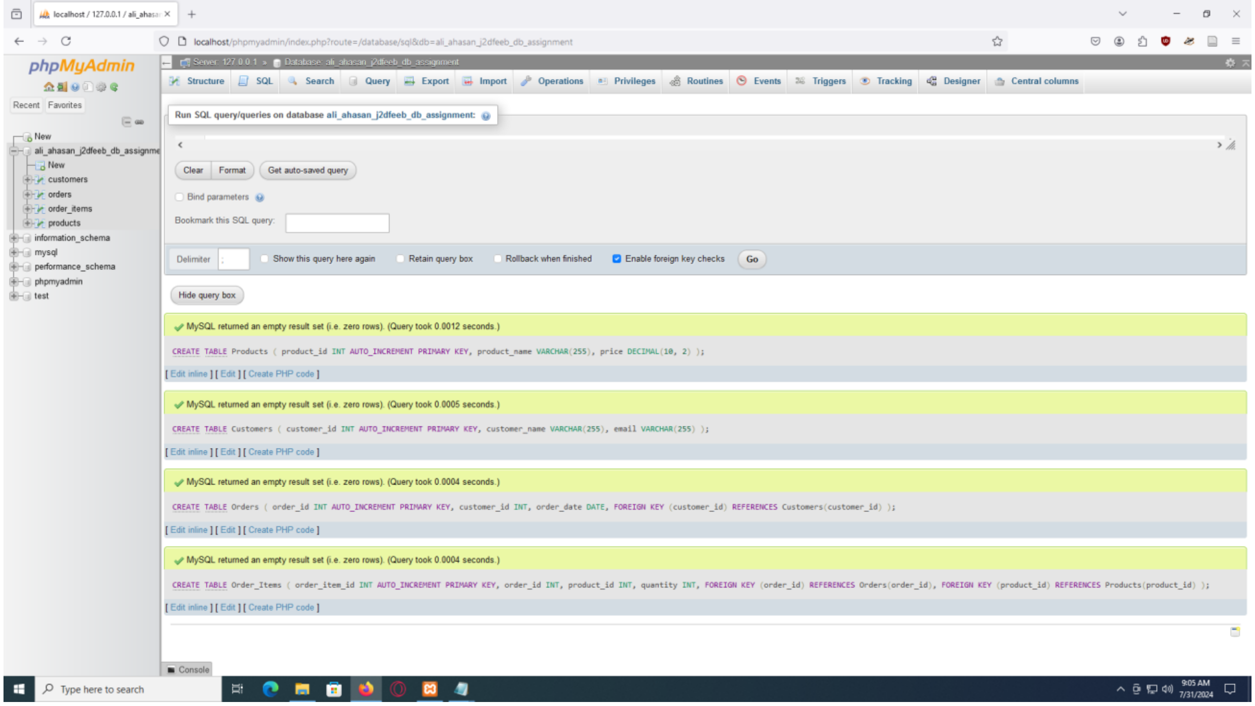
Output:

1



```
✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0012 seconds.)
CREATE TABLE Products ( product_id INT AUTO_INCREMENT PRIMARY KEY, product_name VARCHAR(255), price DECIMAL(10, 2) );
[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0005 seconds.)
CREATE TABLE Customers ( customer_id INT AUTO_INCREMENT PRIMARY KEY, customer_name VARCHAR(255), email VARCHAR(255) );
[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)
CREATE TABLE Orders ( order_id INT AUTO_INCREMENT PRIMARY KEY, customer_id INT, order_date DATE, FOREIGN KEY (customer_id) REFERENCES Customers(customer_id) );
[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)
CREATE TABLE Order_Items ( order_item_id INT AUTO_INCREMENT PRIMARY KEY, order_id INT, product_id INT, quantity INT, FOREIGN KEY (order_id) REFERENCES Orders(order_id), FOREIGN KEY (product_id) REFERENCES Products(product_id) );
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

2



Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:

□ Bind parameters

Bookmark this SQL query:

Delimiter ; □ Show this query here again □ Retain query box □ Rollback when finished ☑ Enable foreign key checks

```
✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0012 seconds.)
CREATE TABLE Products ( product_id INT AUTO_INCREMENT PRIMARY KEY, product_name VARCHAR(255), price DECIMAL(10, 2) );
[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0005 seconds.)
CREATE TABLE Customers ( customer_id INT AUTO_INCREMENT PRIMARY KEY, customer_name VARCHAR(255), email VARCHAR(255) );
[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)
CREATE TABLE Orders ( order_id INT AUTO_INCREMENT PRIMARY KEY, customer_id INT, order_date DATE, FOREIGN KEY (customer_id) REFERENCES Customers(customer_id) );
[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0004 seconds.)
CREATE TABLE Order_Items ( order_item_id INT AUTO_INCREMENT PRIMARY KEY, order_id INT, product_id INT, quantity INT, FOREIGN KEY (order_id) REFERENCES Orders(order_id), FOREIGN KEY (product_id) REFERENCES Products(product_id) );
[ Edit inline ] [ Edit ] [ Create PHP code ]
```

3

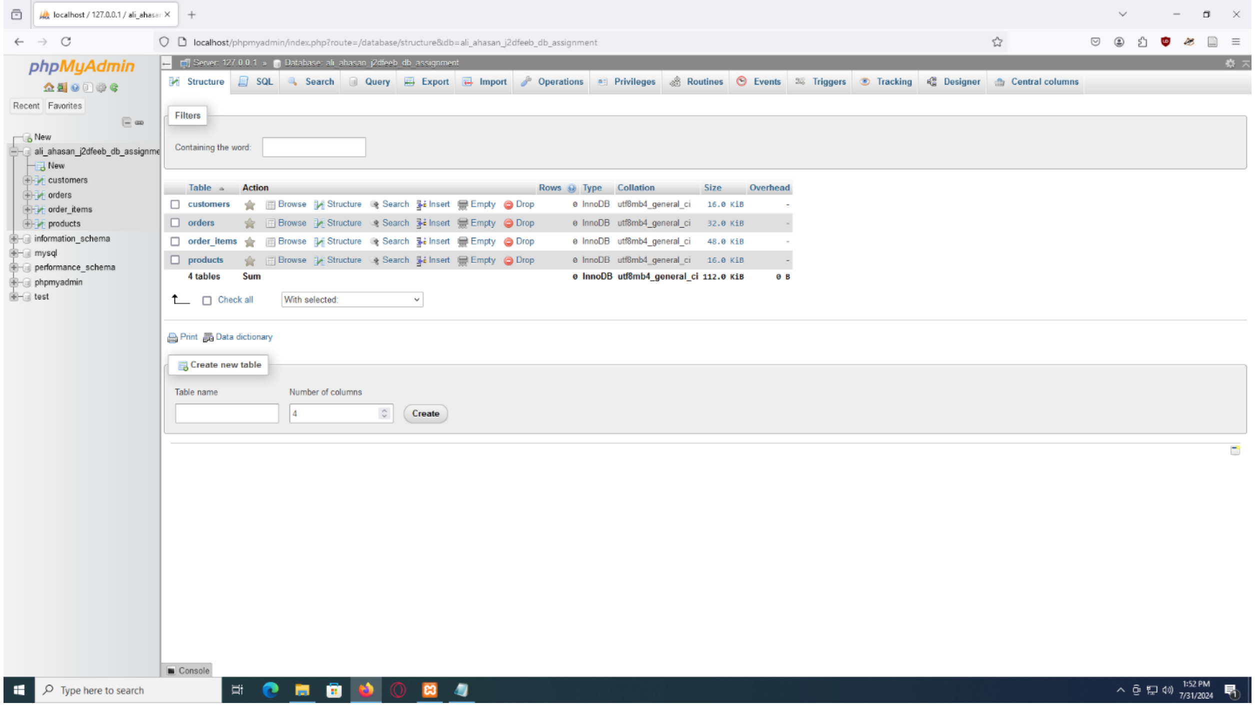


	Table	Action	Rows	Type	Collation	Size	Overhead
□	customers	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
□	orders	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	32.0 KiB	-
□	order_items	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	48.0 KiB	-
□	products	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	4 tables	Sum	0	InnoDB	utf8mb4_general_ci	112.0 KiB	0 B

Table Data Insertion:

Using DML (Data Manipulation Language) to insert the data.

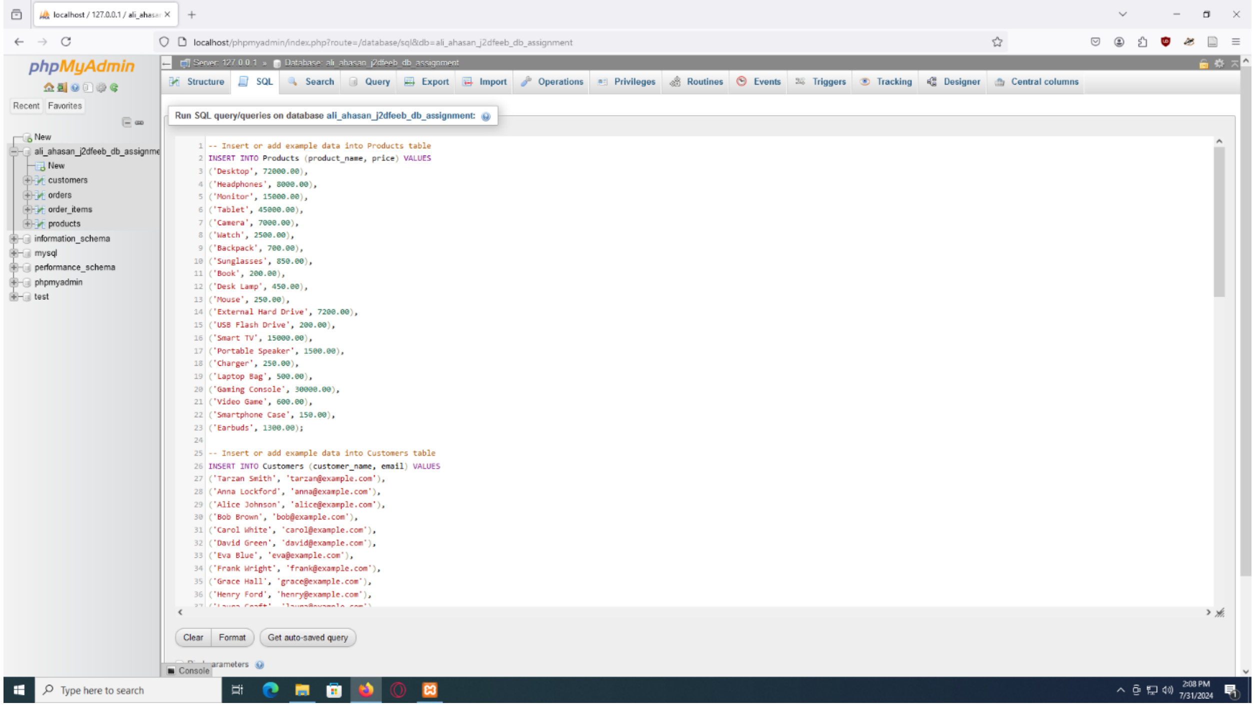
Code:

```
-- Insert or add example data into Products table
INSERT INTO Products (product_name, price) VALUES
('Desktop', 72000.00),
('Headphones', 8000.00),
('Monitor', 15000.00),
('Tablet', 45000.00),
('Camera', 7000.00),
('Watch', 2500.00),
('Backpack', 700.00),
('Sunglasses', 850.00),
('Book', 200.00),
('Desk Lamp', 450.00),
('Mouse', 250.00),
('External Hard Drive', 7200.00),
('USB Flash Drive', 200.00),
('Smart TV', 15000.00),
('Portable Speaker', 1500.00),
('Charger', 250.00),
('Laptop Bag', 500.00),
('Gaming Console', 30000.00),
('Video Game', 600.00),
('Smartphone Case', 150.00),
('Earbuds', 1300.00);

-- Insert or add example data into Customers table
INSERT INTO Customers (customer_name, email) VALUES
```

```
('Tarzan Smith', 'tarzan@example.com'),
('Anna Lockford', 'anna@example.com'),
('Alice Johnson', 'alice@example.com'),
('Bob Brown', 'bob@example.com'),
('Carol White', 'carol@example.com'),
('David Green', 'david@example.com'),
('Eva Blue', 'eva@example.com'),
('Frank Wright', 'frank@example.com'),
('Grace Hall', 'grace@example.com'),
('Henry Ford', 'henry@example.com'),
('Laura Craft', 'laura@example.com'),
('Omar Little', 'omar@example.com'),
('Terry Crews', 'terry@example.com'),
('Nina Williams', 'nina@example.com'),
('Larry Page', 'larry@example.com'),
('Sergey Brin', 'sergey@example.com'),
('Tim Cook', 'tim@example.com'),
('Sundar Pichai', 'sundar@example.com'),
('Elon Musk', 'elon@example.com'),
('Jeff Bezos', 'jeff@example.com');

-- Insert or add example data into Orders table
INSERT INTO Orders (customer_id, order_date) VALUES
(1, '2024-06-11'),
(2, '2024-06-22'),
(3, '2023-07-03'),
(4, '2023-07-04'),
(5, '2023-07-05'),
(6, '2023-07-06'),
(7, '2023-07-07'),
(8, '2023-07-08'),
(9, '2023-07-09'),
(10, '2023-07-10'),
(11, '2023-08-01'),
(12, '2023-08-02'),
(13, '2023-08-03'),
(14, '2023-08-04'),
(15, '2023-08-05'),
(16, '2023-08-06'),
(17, '2023-08-07'),
(18, '2023-08-08'),
(19, '2023-08-09'),
(20, '2023-08-10');


-- Insert or add example data into Order_Items table
INSERT INTO Order_Items (order_id, product_id, quantity) VALUES
(1, 1, 1),
(1, 3, 2),
(2, 2, 1),
(3, 4, 1),
(4, 5, 2),
(5, 6, 3),
(6, 7, 1),
(7, 8, 2),
(8, 9, 1),
(9, 1, 1),
(10, 2, 2),
(10, 3, 3),
(10, 14, 1),
(10, 15, 2),
(11, 11, 2),
(12, 12, 3),
(13, 13, 1),
(14, 14, 1),
(15, 15, 4),
(16, 16, 1),
(17, 17, 1),
(18, 18, 2),
(19, 19, 3),
(20, 20, 1),
(11, 12, 1),
(12, 11, 2),
(13, 13, 3),
(14, 11, 1),
(15, 18, 2),
(16, 16, 3),
(17, 18, 2),
(18, 18, 1),
(19, 9, 1),
(20, 18, 2);
```

Input:

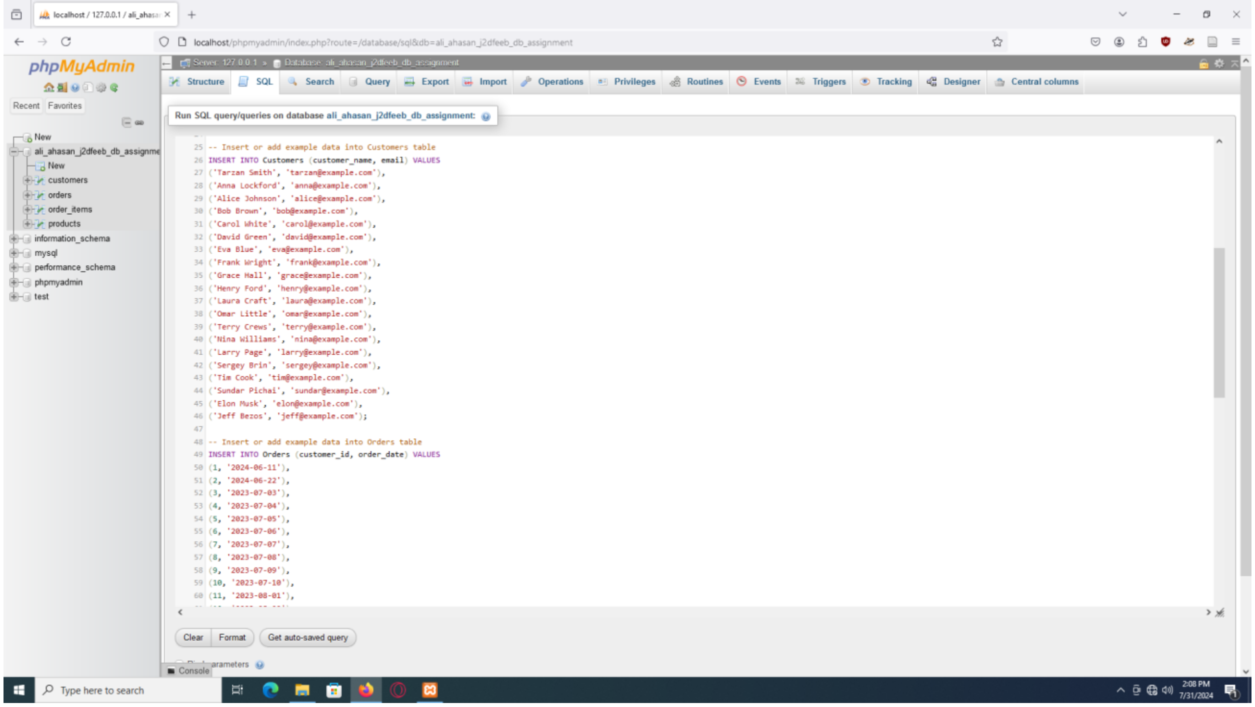
1



Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:

```
-- Insert or add example data into Products table
INSERT INTO Products (product_name, price) VALUES
('Desktop', 72000.00),
('Headphones', 8000.00),
('Monitor', 15000.00),
('Tablet', 45000.00),
('Camera', 7000.00),
('Watch', 2500.00),
('Backpack', 700.00),
('Sunglasses', 850.00),
('Book', 200.00),
('Desk Lamp', 450.00),
('Mouse', 250.00),
('External Hard Drive', 7200.00),
('USB Flash Drive', 200.00),
('Smart TV', 15000.00),
('Portable Speaker', 1500.00),
('Charger', 250.00),
('Laptop Bag', 500.00),
('Gaming Console', 30000.00),
('Video Game', 600.00),
('Smartphone Case', 150.00),
('Earbuds', 1300.00);

-- Insert or add example data into Customers table
INSERT INTO Customers (customer_name, email) VALUES
('Tarzan Smith', 'tarzan@example.com'),
('Anna Lockford', 'anna@example.com'),
('Alice Johnson', 'alice@example.com'),
('Bob Brown', 'bob@example.com'),
('Carol White', 'carol@example.com'),
('David Green', 'david@example.com'),
('Eva Blue', 'eva@example.com'),
('Frank Wright', 'frank@example.com'),
('Grace Hall', 'grace@example.com'),
('Henry Ford', 'henry@example.com'),
```

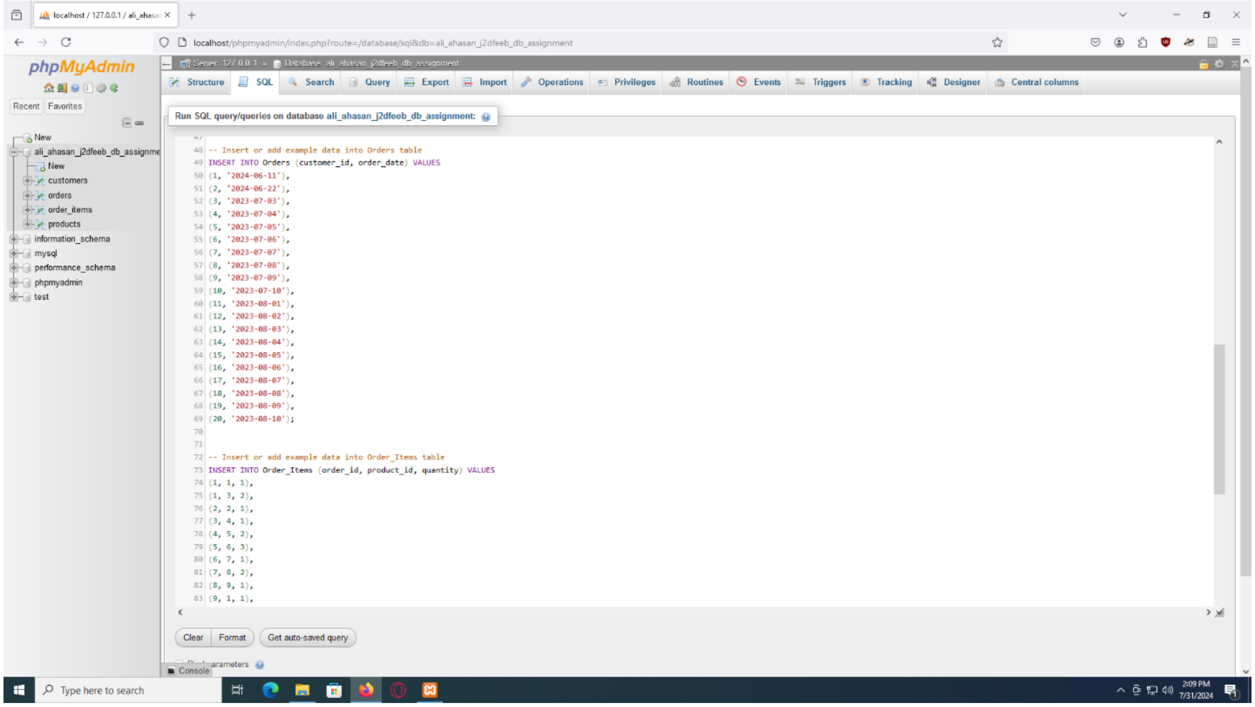
2



Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:

```sql
-- Insert or add example data into Customers table
INSERT INTO Customers (customer_name, email) VALUES
('Tarzan Smith', 'tarzan@example.com'),
('Anna Lockford', 'anna@example.com'),
('Alice Johnson', 'alice@example.com'),
('Bob Brown', 'bob@example.com'),
('Carol White', 'carol@example.com'),
('David Green', 'david@example.com'),
('Eva Blue', 'eva@example.com'),
('Frank Wright', 'frank@example.com'),
('Grace Hall', 'grace@example.com'),
('Henry Ford', 'henry@example.com'),
('Laura Craft', 'laura@example.com'),
('Omar Little', 'omar@example.com'),
('Terry Crews', 'terry@example.com'),
('Nina Williams', 'nina@example.com'),
('Larry Page', 'larry@example.com'),
('Sergey Brin', 'sergey@example.com'),
('Tim Cook', 'tim@example.com'),
('Sundar Pichai', 'sundar@example.com'),
('Elon Musk', 'elon@example.com'),
('Jeff Bezos', 'jeff@example.com');

-- Insert or add example data into Orders table
INSERT INTO Orders (customer_id, order_date) VALUES
(1, '2024-06-11'),
(2, '2024-06-22'),
(3, '2023-07-03'),
(4, '2023-07-04'),
(5, '2023-07-05'),
(6, '2023-07-06'),
(7, '2023-07-07'),
(8, '2023-07-08'),
(9, '2023-07-09'),
(10, '2023-07-10'),
(11, '2023-08-01'),
```

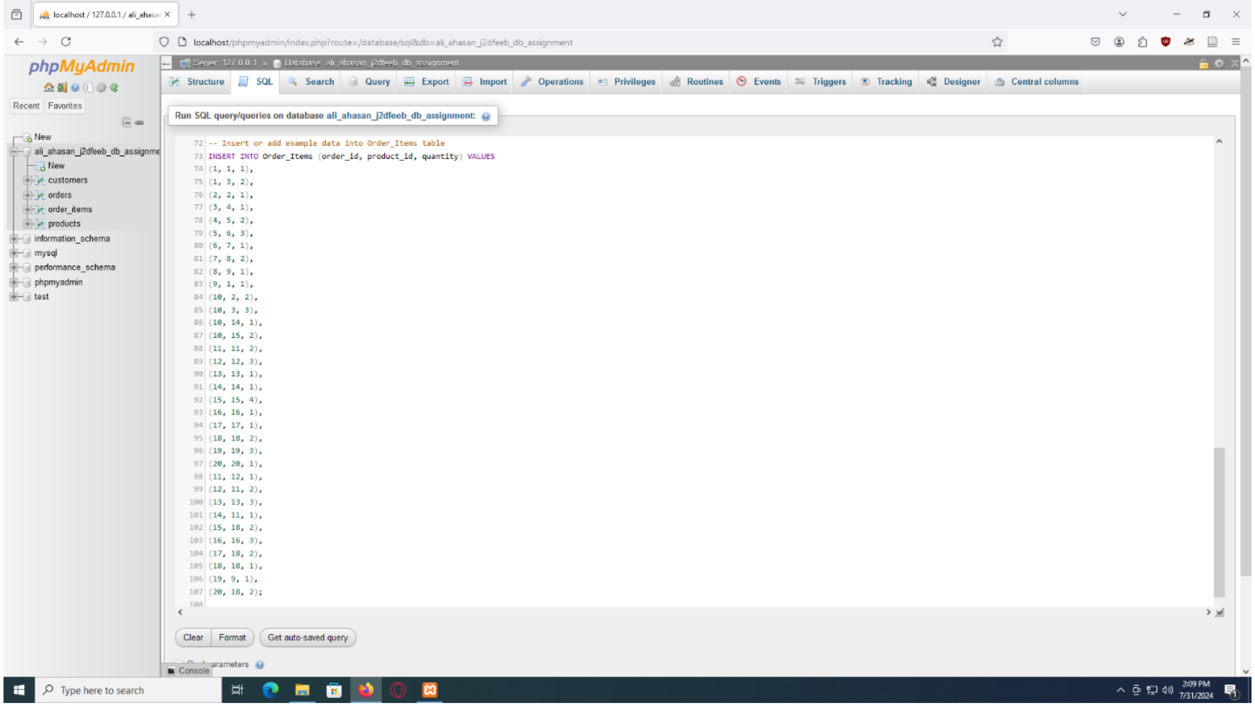
3



Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:

```sql
-- Insert or add example data into Orders table
INSERT INTO Orders (customer_id, order_date) VALUES
(1, '2024-06-11'),
(2, '2024-06-22'),
(3, '2023-07-03'),
(4, '2023-07-04'),
(5, '2023-07-05'),
(6, '2023-07-06'),
(7, '2023-07-07'),
(8, '2023-07-08'),
(9, '2023-07-09'),
(10, '2023-07-10'),
(11, '2023-08-01'),
(12, '2023-08-02'),
(13, '2023-08-03'),
(14, '2023-08-04'),
(15, '2023-08-05'),
(16, '2023-08-06'),
(17, '2023-08-07'),
(18, '2023-08-08'),
(19, '2023-08-09'),
(20, '2023-08-10');


-- Insert or add example data into Order_Items table
INSERT INTO Order_Items (order_id, product_id, quantity) VALUES
(1, 1, 1),
(1, 3, 2),
(2, 2, 1),
(3, 4, 1),
(4, 5, 2),
(5, 6, 3),
(6, 7, 1),
(7, 8, 2),
(8, 9, 1),
(9, 1, 1),
```

4

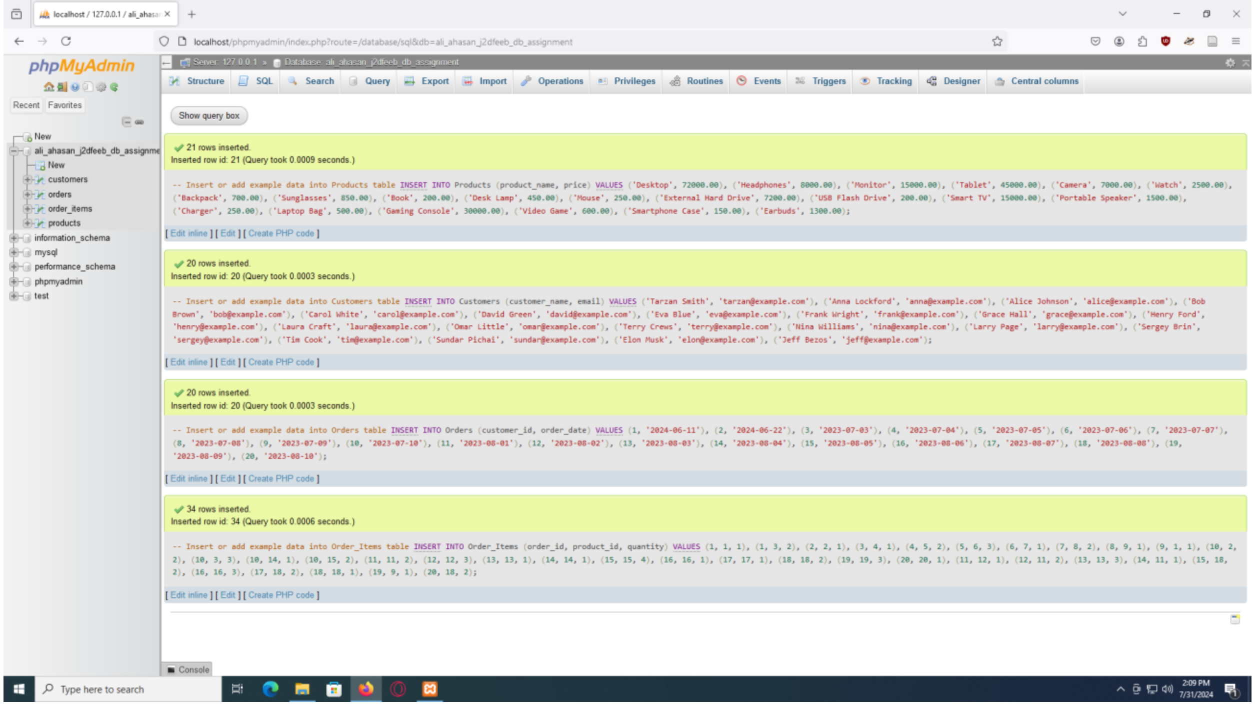


Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:

```
-- Insert or add example data into Order_Items table
INSERT INTO Order_Items (order_id, product_id, quantity) VALUES
(1, 1, 1),
(1, 3, 2),
(2, 2, 1),
(3, 4, 1),
(4, 5, 2),
(5, 6, 3),
(6, 7, 1),
(7, 8, 2),
(8, 9, 1),
(9, 1, 1),
(10, 2, 2),
(10, 3, 3),
(10, 14, 1),
(10, 15, 2),
(11, 11, 2),
(12, 12, 3),
(13, 13, 1),
(14, 14, 1),
(15, 15, 4),
(16, 16, 1),
(17, 17, 1),
(18, 18, 2),
(19, 19, 3),
(20, 20, 1),
(11, 12, 1),
(12, 11, 2),
(13, 13, 3),
(14, 11, 1),
(15, 18, 2),
(16, 16, 3),
(17, 18, 2),
(18, 18, 1),
(19, 9, 1),
(20, 18, 2);
```

Output:

1


localhost / 127.0.0.1 / ali_ahasan
localhost/phpmyadmin/index.php?route=/database/sql&db=ali_ahasan_j2dfeeb_db_assignment
phpMyAdmin
Recent Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer Central columns
Show query box
Console
Type here to search
2:09 PM
7/31/2024

✓ 21 rows inserted.
Inserted row id: 21 (Query took 0.0009 seconds.)

```
-- Insert or add example data into Products table INSERT INTO Products (product_name, price) VALUES ('Desktop', 72000.00), ('Headphones', 8000.00), ('Monitor', 15000.00), ('Tablet', 45000.00), ('Camera', 7000.00), ('Watch', 2500.00),
('Backpack', 700.00), ('Sunglasses', 850.00), ('Book', 200.00), ('Desk Lamp', 450.00), ('Mouse', 250.00), ('External Hard Drive', 7200.00), ('USB Flash Drive', 200.00), ('Smart TV', 15000.00), ('Portable Speaker', 1500.00),
('Charger', 250.00), ('Laptop Bag', 500.00), ('Gaming Console', 30000.00), ('Video Game', 600.00), ('Smartphone Case', 150.00), ('Earbuds', 1300.00);
```

[Edit inline] [Edit] [Create PHP code]

✓ 20 rows inserted.
Inserted row id: 20 (Query took 0.0003 seconds.)

```
-- Insert or add example data into Customers table INSERT INTO Customers (customer_name, email) VALUES ('Tarzan Smith', 'tarzan@example.com'), ('Anna Lockford', 'anna@example.com'), ('Alice Johnson', 'alice@example.com'), ('Bob
Brown', 'bob@example.com'), ('Carol White', 'carol@example.com'), ('David Green', 'david@example.com'), ('Eva Blue', 'eva@example.com'), ('Frank Wright', 'frank@example.com'), ('Grace Hall', 'grace@example.com'), ('Henry Ford',
'henry@example.com'), ('Laura Craft', 'laura@example.com'), ('Omar Little', 'omar@example.com'), ('Terry Crews', 'terry@example.com'), ('Nina Williams', 'nina@example.com'), ('Larry Page', 'larry@example.com'), ('Sergey Brin',
'sergey@example.com'), ('Tim Cook', 'tim@example.com'), ('Sundar Pichai', 'sundar@example.com'), ('Elon Musk', 'elon@example.com'), ('Jeff Bezos', 'jeff@example.com');
```

[Edit inline] [Edit] [Create PHP code]

✓ 20 rows inserted.
Inserted row id: 20 (Query took 0.0003 seconds.)

```
-- Insert or add example data into Orders table INSERT INTO Orders (customer_id, order_date) VALUES (1, '2024-06-11'), (2, '2024-06-22'), (3, '2023-07-03'), (4, '2023-07-04'), (5, '2023-07-05'), (6, '2023-07-06'), (7, '2023-07-07'),
(8, '2023-07-08'), (9, '2023-07-09'), (10, '2023-07-10'), (11, '2023-08-01'), (12, '2023-08-02'), (13, '2023-08-03'), (14, '2023-08-04'), (15, '2023-08-05'), (16, '2023-08-06'), (17, '2023-08-07'), (18, '2023-08-08'), (19,
'2023-08-09'), (20, '2023-08-10');
```

[Edit inline] [Edit] [Create PHP code]

✓ 34 rows inserted.
Inserted row id: 34 (Query took 0.0006 seconds.)

```
-- Insert or add example data into Order_Items table INSERT INTO Order_Items (order_id, product_id, quantity) VALUES (1, 1, 1), (1, 3, 2), (2, 2, 1), (3, 4, 1), (4, 5, 2), (5, 6, 3), (6, 7, 1), (7, 8, 2), (8, 9, 1), (9, 1, 1), (10, 2,
2), (10, 3, 3), (10, 14, 1), (10, 15, 2), (11, 11, 2), (12, 12, 3), (13, 13, 1), (14, 14, 1), (15, 15, 4), (16, 16, 1), (17, 17, 1), (18, 18, 2), (19, 19, 3), (20, 20, 1), (11, 12, 1), (12, 11, 2), (13, 13, 3), (14, 11, 1), (15, 18,
2), (16, 16, 3), (17, 18, 2), (18, 18, 1), (19, 9, 1), (20, 18, 2);
```

[Edit inline] [Edit] [Create PHP code]

2

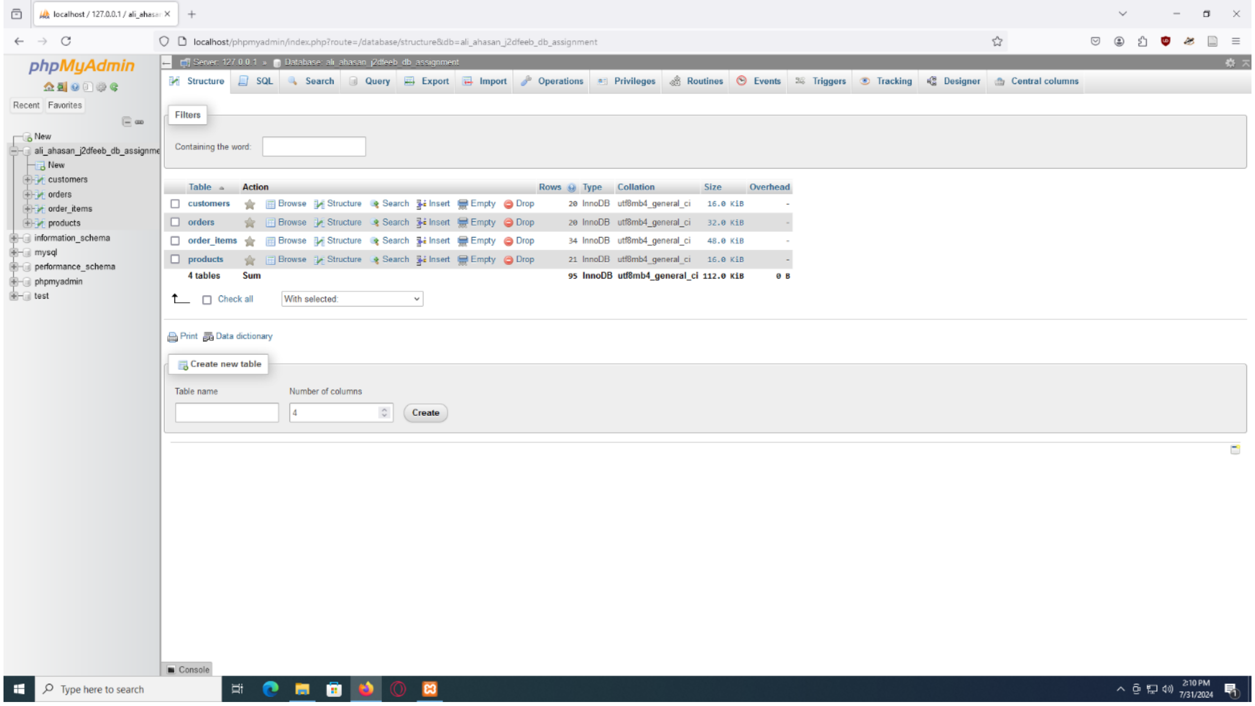
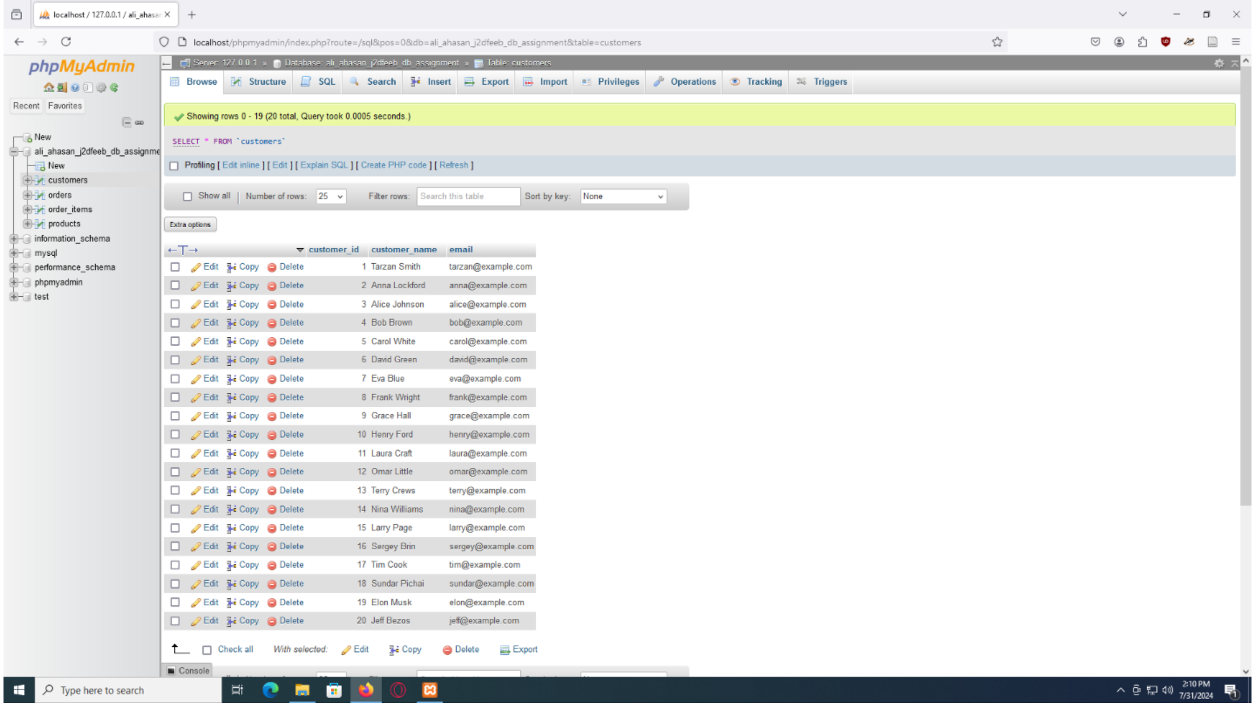

localhost / 127.0.0.1 / ali_ahasan
localhost/phpmyadmin/index.php?route=/database/structure&db=ali_ahasan_j2dfeeb_db_assignment
phpMyAdmin
Recent Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer Central columns
Filters
Containing the word:
Check all
With selected:
Print Data dictionary
Create new table
Table name
Number of columns
Create
Console
Type here to search
2:10 PM
7/31/2024

	Table	Action	Rows	Type	Collation	Size	Overhead
□	customers	Browse Structure Search Insert Empty Drop	20	InnoDB	utf8mb4_general_ci	16.0 KiB	-
□	orders	Browse Structure Search Insert Empty Drop	20	InnoDB	utf8mb4_general_ci	32.0 KiB	-
□	order_items	Browse Structure Search Insert Empty Drop	34	InnoDB	utf8mb4_general_ci	48.0 KiB	-
□	products	Browse Structure Search Insert Empty Drop	21	InnoDB	utf8mb4_general_ci	16.0 KiB	-
	4 tables	Sum	95	InnoDB	utf8mb4_general_ci	112.0 KiB	0 B

Number of columns: 4

3



localhost/phpmyadmin/index.php?route=/sql&pos=0&db=ali_ahasan_j2dfeeb_db_assignment&table=customers

Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment » Table: customers

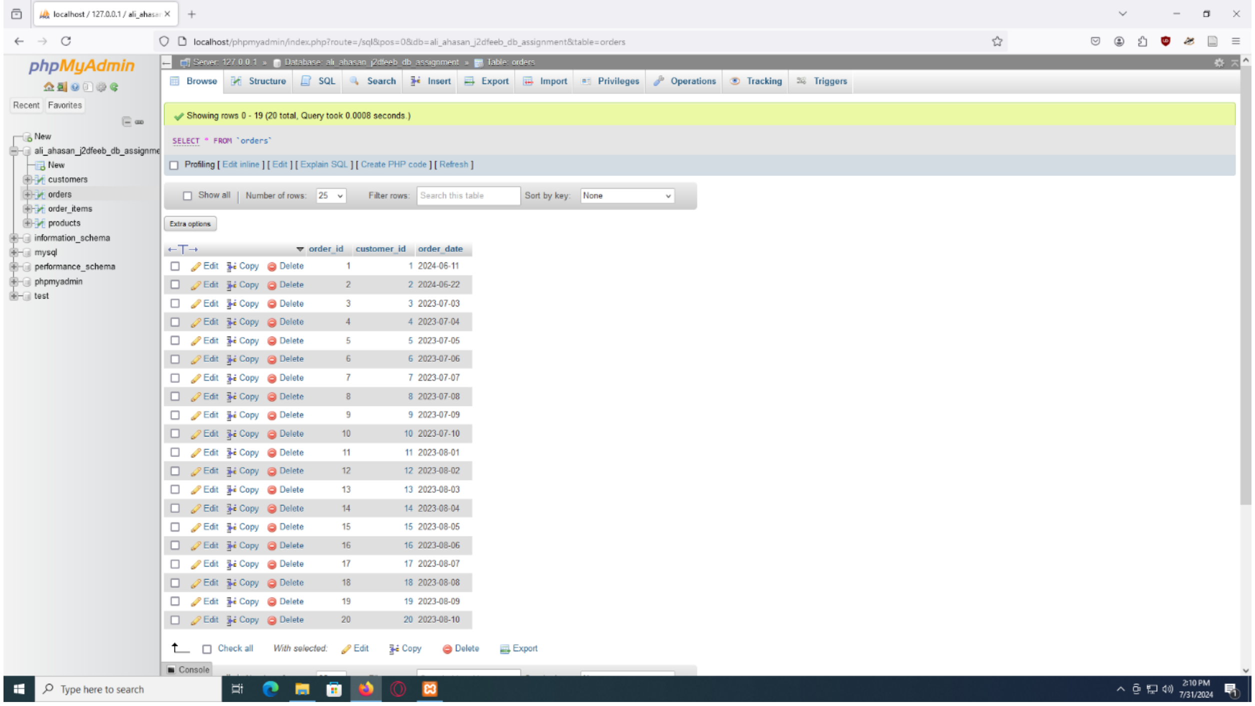
✓ Showing rows 0 - 19 (20 total, Query took 0.0005 seconds.)

```
SELECT * FROM `customers`
```

customer_id	customer_name	email
1	Tarzan Smith	tarzan@example.com
2	Anna Lockford	anna@example.com
3	Alice Johnson	alice@example.com
4	Bob Brown	bob@example.com
5	Carol White	carol@example.com
6	David Green	david@example.com
7	Eva Blue	eva@example.com
8	Frank Wright	frank@example.com
9	Grace Hall	grace@example.com
10	Henry Ford	henry@example.com
11	Laura Craft	laura@example.com
12	Omar Little	omar@example.com
13	Terry Crews	terry@example.com
14	Nina Williams	nina@example.com
15	Larry Page	larry@example.com
16	Sergey Brin	sergey@example.com
17	Tim Cook	tim@example.com
18	Sundar Pichai	sundar@example.com
19	Elon Musk	elon@example.com
20	Jeff Bezos	jeff@example.com

localhost / 127.0.0.1 / ali_ahasan
phpMyAdmin
Recent Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options
Edit Copy Delete
Check all With selected: Edit Copy Delete Export
Console
Type here to search
2:10 PM
7/31/2024

4



localhost/phpmyadmin/index.php?route=/sql&pos=0&db=ali_ahasan_j2dfeeb_db_assignment&table=orders

Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment » Table: orders

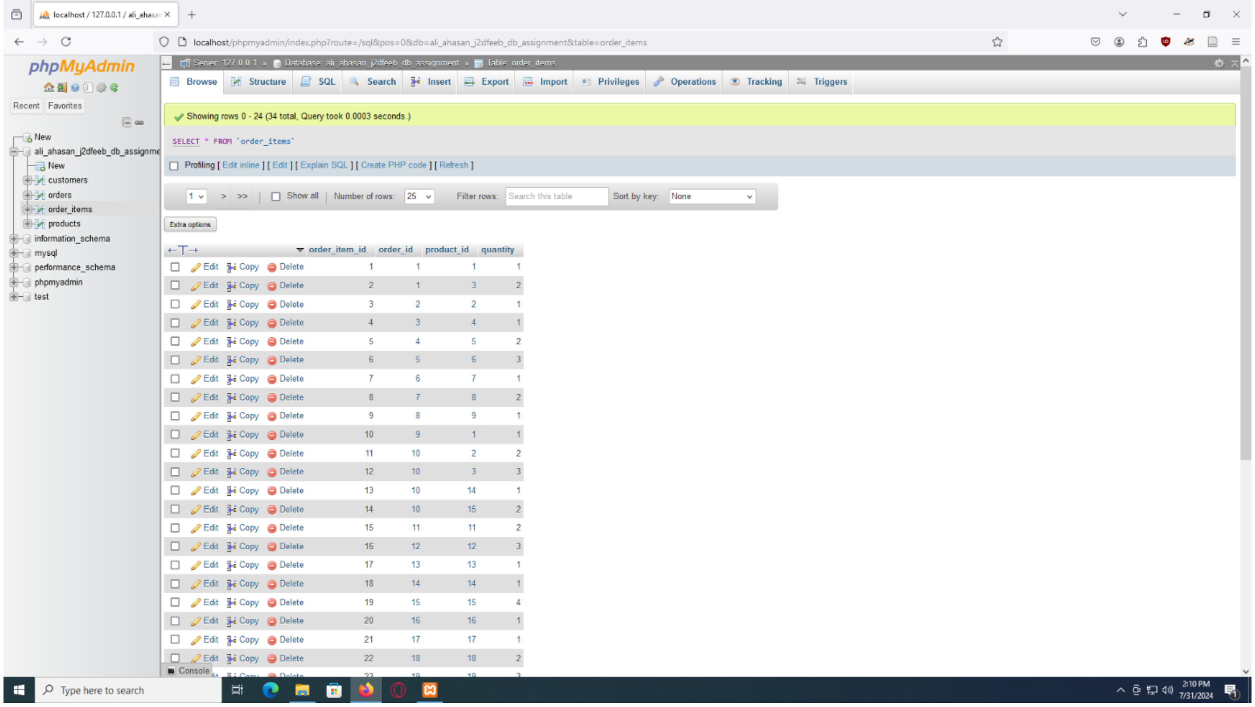
✓ Showing rows 0 - 19 (20 total, Query took 0.0008 seconds.)

```
SELECT * FROM `orders`
```

order_id	customer_id	order_date
1	1	2024-06-11
2	2	2024-06-22
3	3	2023-07-03
4	4	2023-07-04
5	5	2023-07-05
6	6	2023-07-06
7	7	2023-07-07
8	8	2023-07-08
9	9	2023-07-09
10	10	2023-07-10
11	11	2023-08-01
12	12	2023-08-02
13	13	2023-08-03
14	14	2023-08-04
15	15	2023-08-05
16	16	2023-08-06
17	17	2023-08-07
18	18	2023-08-08
19	19	2023-08-09
20	20	2023-08-10

localhost / 127.0.0.1 / ali_ahasan
phpMyAdmin
Recent Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options
Edit Copy Delete
Check all With selected: Edit Copy Delete Export
Console
Type here to search
2:10 PM
7/31/2024

5



localhost / 127.0.0.1 / ali_ahasan
localhost/phpmyadmin/index.php?route=/sql&pos=0&db=ali_ahasan_j2dfeeb_db_assignment&table=order_items
Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment » Table: order_items
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
phpMyAdmin
Recent Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Extra options
Console
Type here to search
2:10 PM
7/31/2024

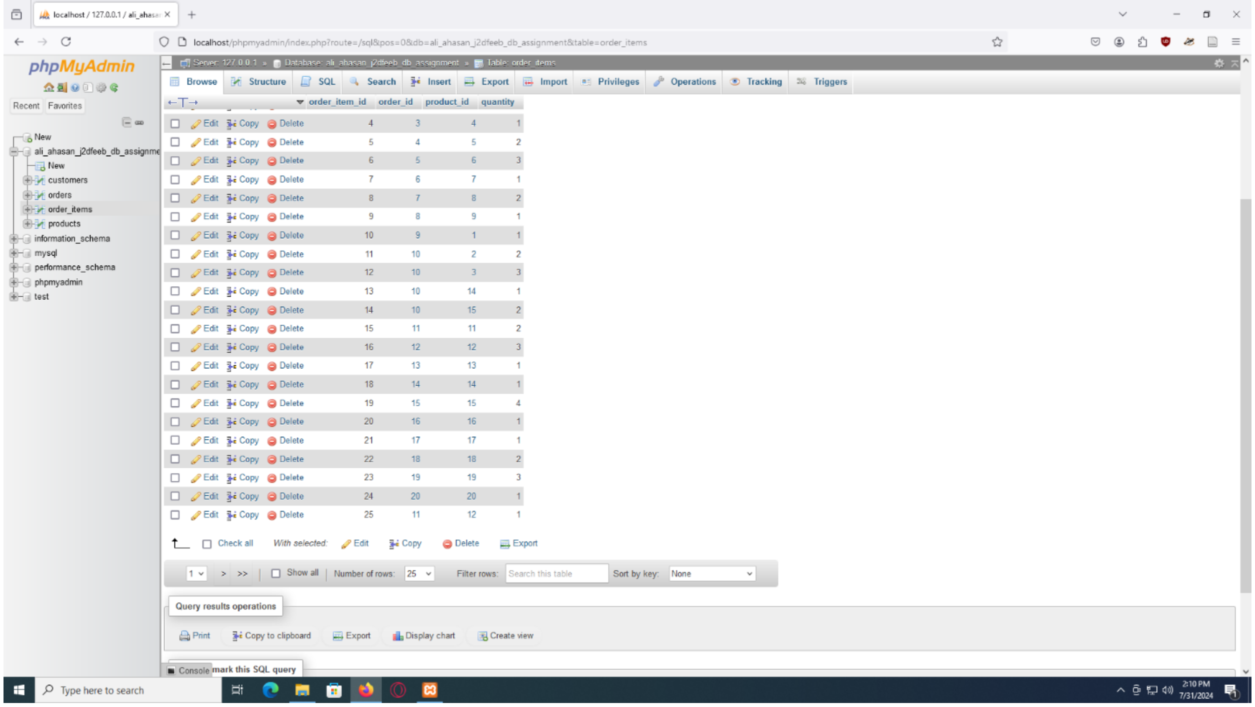
Showing rows 0 - 24 (34 total, Query took 0.0003 seconds.)

```
SELECT * FROM `order_items`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | □ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

	order_item_id	order_id	product_id	quantity
□ Edit Copy Delete	1	1	1	1
□ Edit Copy Delete	2	1	3	2
□ Edit Copy Delete	3	2	2	1
□ Edit Copy Delete	4	3	4	1
□ Edit Copy Delete	5	4	5	2
□ Edit Copy Delete	6	5	6	3
□ Edit Copy Delete	7	6	7	1
□ Edit Copy Delete	8	7	8	2
□ Edit Copy Delete	9	8	9	1
□ Edit Copy Delete	10	9	1	1
□ Edit Copy Delete	11	10	2	2
□ Edit Copy Delete	12	10	3	3
□ Edit Copy Delete	13	10	14	1
□ Edit Copy Delete	14	10	15	2
□ Edit Copy Delete	15	11	11	2
□ Edit Copy Delete	16	12	12	3
□ Edit Copy Delete	17	13	13	1
□ Edit Copy Delete	18	14	14	1
□ Edit Copy Delete	19	15	15	4
□ Edit Copy Delete	20	16	16	1
□ Edit Copy Delete	21	17	17	1
□ Edit Copy Delete	22	18	18	2



localhost / 127.0.0.1 / ali_ahasan
localhost/phpmyadmin/index.php?route=/sql&pos=0&db=ali_ahasan_j2dfeeb_db_assignment&table=order_items
Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment » Table: order_items
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
phpMyAdmin
Recent Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Console
mark this SQL query
Type here to search
2:10 PM
7/31/2024

	order_item_id	order_id	product_id	quantity
□ Edit Copy Delete	4	3	4	1
□ Edit Copy Delete	5	4	5	2
□ Edit Copy Delete	6	5	6	3
□ Edit Copy Delete	7	6	7	1
□ Edit Copy Delete	8	7	8	2
□ Edit Copy Delete	9	8	9	1
□ Edit Copy Delete	10	9	1	1
□ Edit Copy Delete	11	10	2	2
□ Edit Copy Delete	12	10	3	3
□ Edit Copy Delete	13	10	14	1
□ Edit Copy Delete	14	10	15	2
□ Edit Copy Delete	15	11	11	2
□ Edit Copy Delete	16	12	12	3
□ Edit Copy Delete	17	13	13	1
□ Edit Copy Delete	18	14	14	1
□ Edit Copy Delete	19	15	15	4
□ Edit Copy Delete	20	16	16	1
□ Edit Copy Delete	21	17	17	1
□ Edit Copy Delete	22	18	18	2
□ Edit Copy Delete	23	19	19	3
□ Edit Copy Delete	24	20	20	1
□ Edit Copy Delete	25	11	12	1

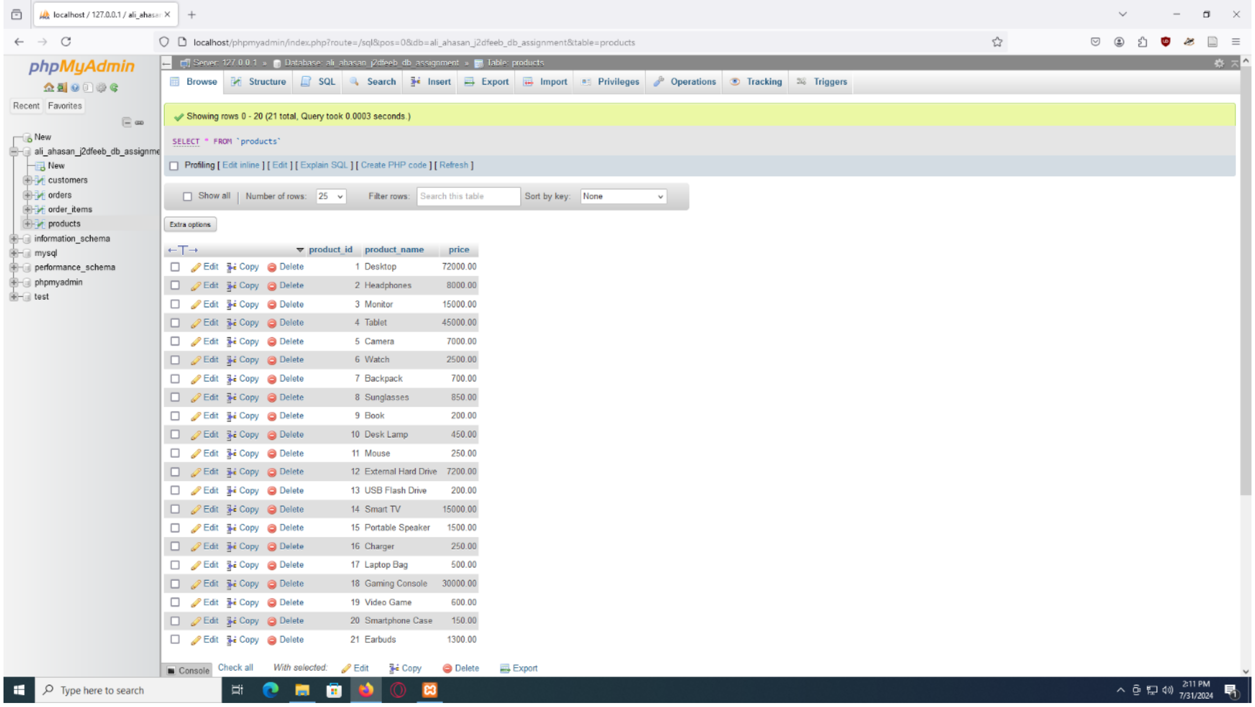
□ Check all *With selected:* Edit Copy Delete Export

1 > >> | □ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

6



localhost/phpmyadmin/index.php?route=/sql&pos=0&db=ali_ahasan_j2dfeeb_db_assignment&table=products
Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment » Table: products
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Operations
Tracking
Triggers

Showing rows 0 - 20 (21 total, Query took 0.0003 seconds.)

```
SELECT * FROM `products`
```

product_id	product_name	price
1	Desktop	72000.00
2	Headphones	8000.00
3	Monitor	15000.00
4	Tablet	45000.00
5	Camera	7000.00
6	Watch	2500.00
7	Backpack	700.00
8	Sunglasses	850.00
9	Book	200.00
10	Desk Lamp	450.00
11	Mouse	250.00
12	External Hard Drive	7200.00
13	USB Flash Drive	200.00
14	Smart TV	15000.00
15	Portable Speaker	1500.00
16	Charger	250.00
17	Laptop Bag	500.00
18	Gaming Console	30000.00
19	Video Game	600.00
20	Smartphone Case	150.00
21	Earbuds	1300.00

Table Data Query:

With the help of query commands, Data query can be done.

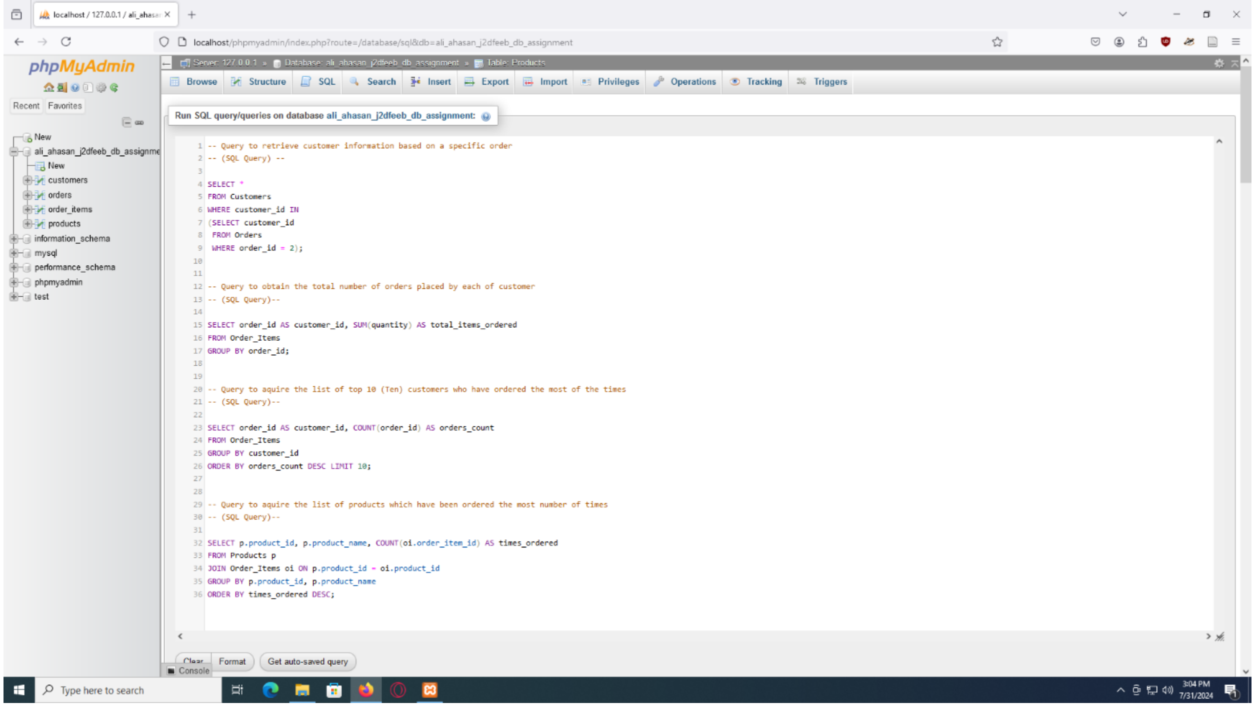
Query code:

```
-- Query to retrieve customer information based on a specific order
-- (SQL Query) --

SELECT *
FROM Customers
WHERE customer_id IN
(SELECT customer_id
 FROM Orders
 WHERE order_id = 2);


-- Query to obtain the total number of orders placed by each of customer
-- (SQL Query)--

SELECT order_id AS customer_id, SUM(quantity) AS total_items_ordered
FROM Order_Items
GROUP BY order_id;


-- Query to aquire the list of top 10 (Ten) customers who have ordered the most of the times
-- (SQL Query)--

SELECT order_id AS customer_id, COUNT(order_id) AS orders_count
FROM Order_Items
```

```
GROUP BY customer_id
ORDER BY orders_count DESC LIMIT 10;


-- Query to aquire the list of products which have been ordered the most number of times
-- (SQL Query)--

SELECT p.product_id, p.product_name, COUNT(oi.order_item_id) AS times_ordered
FROM Products p
JOIN Order_Items oi ON p.product_id = oi.product_id
GROUP BY p.product_id, p.product_name
ORDER BY times_ordered DESC;
```

Input:

1



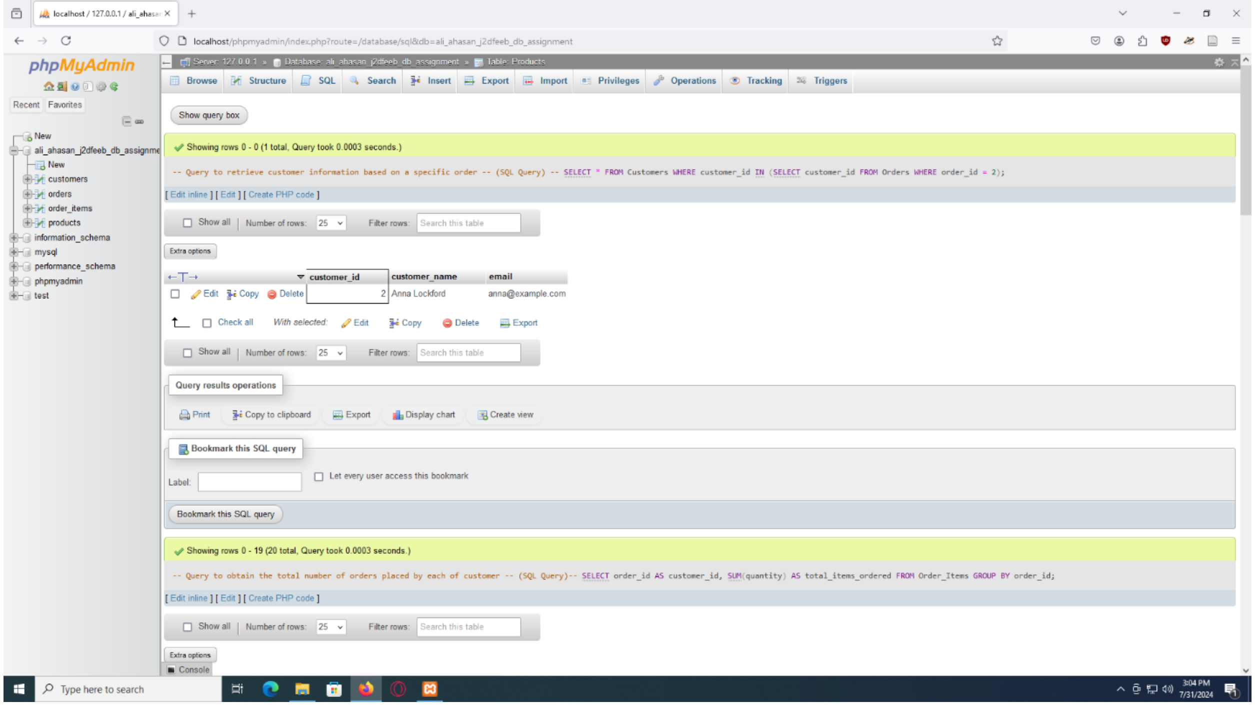
localhost / 127.0.0.1 / ali_ahasa
localhost/phpmyadmin/index.php?route=/database/sql&db=ali_ahasan_j2dfeeb_db_assignment
phpMyAdmin
Recent
Favorites
New
ali_ahasan_j2dfeeb_db_assignme
New
customers
orders
order_items
products
information_schema
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: ali_ahasan_j2dfeeb_db_assignment » Table: Products
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Operations
Tracking
Triggers
Run SQL query/queries on database ali_ahasan_j2dfeeb_db_assignment:
Clear
Format
Get auto-saved query
Console
Type here to search
3:04 PM
7/31/2024

```
-- Query to retrieve customer information based on a specific order
-- (SQL Query) --

SELECT *
FROM Customers
WHERE customer_id IN
(SELECT customer_id
 FROM Orders
 WHERE order_id = 2);


-- Query to obtain the total number of orders placed by each of customer
-- (SQL Query)--

SELECT order_id AS customer_id, SUM(quantity) AS total_items_ordered
FROM Order_Items
GROUP BY order_id;


-- Query to aquire the list of top 10 (Ten) customers who have ordered the most of the times
-- (SQL Query)--

SELECT order_id AS customer_id, COUNT(order_id) AS orders_count
FROM Order_Items
GROUP BY customer_id
ORDER BY orders_count DESC LIMIT 10;


-- Query to aquire the list of products which have been ordered the most number of times
-- (SQL Query)--

SELECT p.product_id, p.product_name, COUNT(oi.order_item_id) AS times_ordered
FROM Products p
JOIN Order_Items oi ON p.product_id = oi.product_id
GROUP BY p.product_id, p.product_name
ORDER BY times_ordered DESC;
```

Output:

1

```
-- Query to retrieve customer information based on a specific order
-- (SQL Query) --

SELECT *
FROM Customers
WHERE customer_id IN
(SELECT customer_id
 FROM Orders
WHERE order_id = 2);
```



Showing rows 0 - 0 (1 total, Query took 0.0003 seconds.)

```
-- Query to retrieve customer information based on a specific order -- (SQL Query) -- SELECT * FROM Customers WHERE customer_id IN (SELECT customer_id FROM Orders WHERE order_id = 2);
```

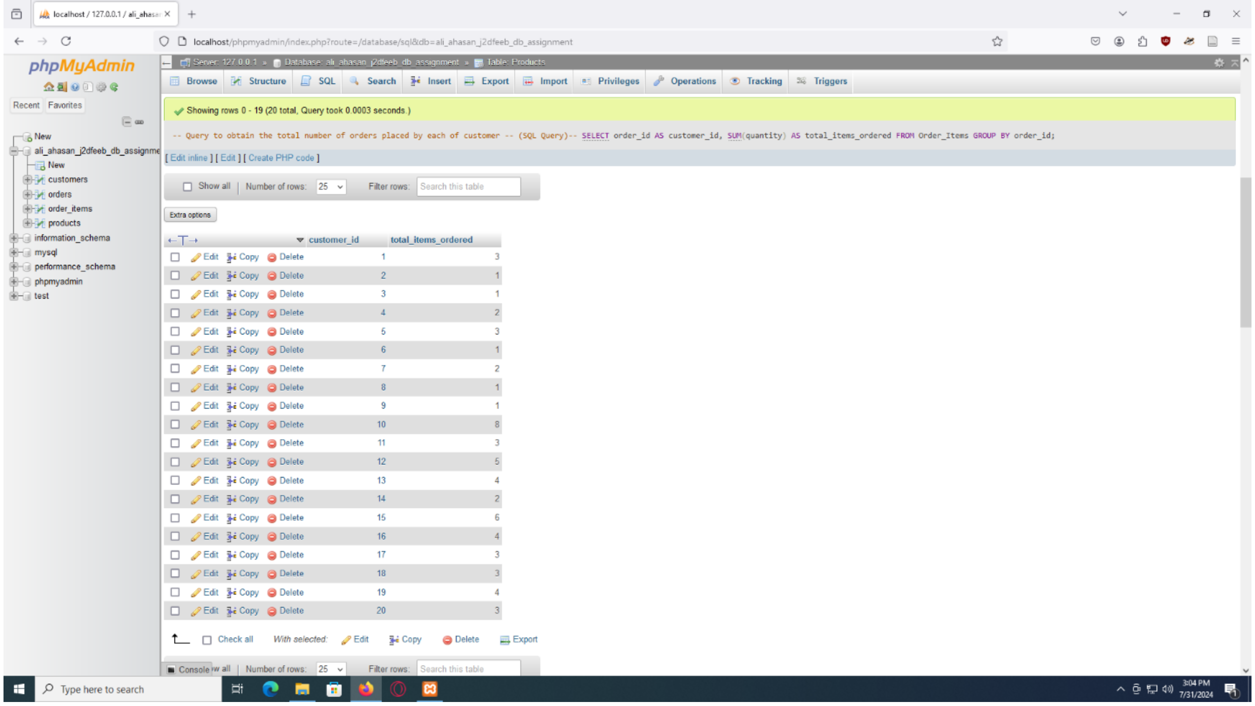
customer_id	customer_name	email
2	Anna Lockford	anna@example.com

Showing rows 0 - 19 (20 total, Query took 0.0003 seconds.)

```
-- Query to obtain the total number of orders placed by each of customer -- (SQL Query)-- SELECT order_id AS customer_id, SUM(quantity) AS total_items_ordered FROM Order_Items GROUP BY order_id;
```

2

```
-- Query to obtain the total number of orders placed by each of customer
-- (SQL Query)--

SELECT order_id AS customer_id, SUM(quantity) AS total_items_ordered
FROM Order_Items
GROUP  BY order_id;
```



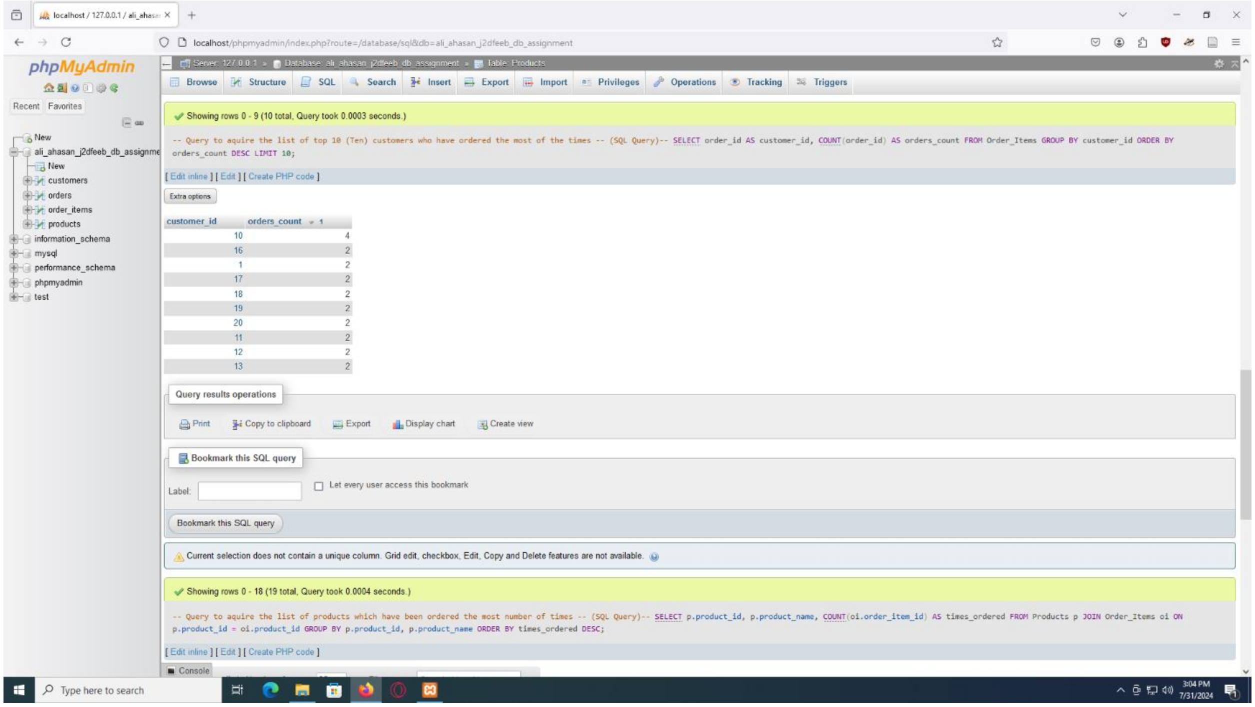
Showing rows 0 - 19 (20 total, Query took 0.0003 seconds.)

```
-- Query to obtain the total number of orders placed by each of customer -- (SQL Query)-- SELECT order_id AS customer_id, SUM(quantity) AS total_items_ordered FROM Order_Items GROUP BY order_id;
```

customer_id	total_items_ordered
1	3
2	1
3	1
4	2
5	3
6	1
7	2
8	1
9	1
10	8
11	3
12	5
13	4
14	2
15	6
16	4
17	3
18	3
19	4
20	3

3

```
-- Query to aquire the list of top 10 (Ten) customers who have ordered the most of the times
-- (SQL Query)--

SELECT order_id AS customer_id, COUNT(order_id) AS orders_count
FROM Order_Items
GROUP BY customer_id
ORDER BY orders_count DESC LIMIT 10;
```



✓ Showing rows 0 - 9 (10 total, Query took 0.0003 seconds.)

```
-- Query to aquire the list of top 10 (Ten) customers who have ordered the most of the times -- (SQL Query)-- SELECT order_id AS customer_id, COUNT(order_id) AS orders_count FROM Order_Items GROUP BY customer_id ORDER BY
orders_count DESC LIMIT 10;
```

customer_id	orders_count
10	4
16	2
1	2
17	2
18	2
19	2
20	2
11	2
12	2
13	2

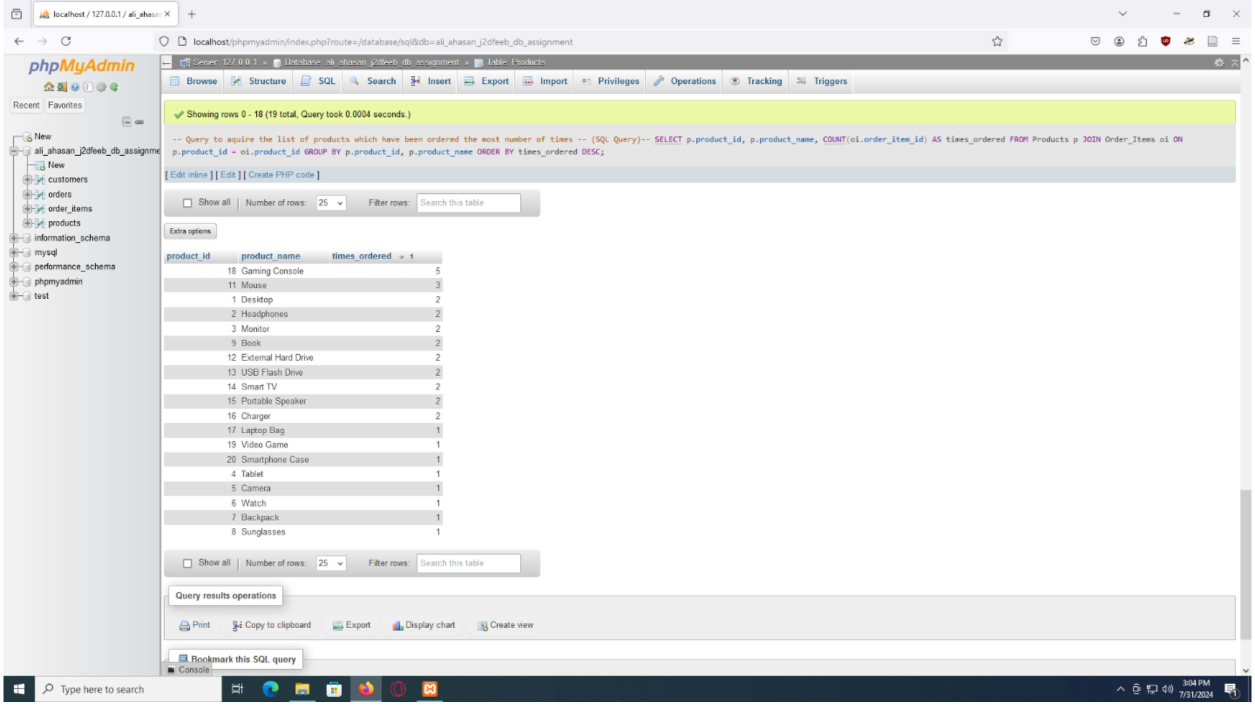
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 18 (19 total, Query took 0.0004 seconds.)

```
-- Query to aquire the list of products which have been ordered the most number of times -- (SQL Query)-- SELECT p.product_id, p.product_name, COUNT(oi.order_item_id) AS times_ordered FROM Products p JOIN Order_Items oi ON
p.product_id = oi.product_id GROUP BY p.product_id, p.product_name ORDER BY times_ordered DESC;
```

4

```
-- Query to aquire the list of products which have been ordered the most number of times
-- (SQL Query)--

SELECT p.product_id, p.product_name, COUNT(oi.order_item_id) AS times_ordered
FROM Products p
JOIN Order_Items oi ON p.product_id = oi.product_id
GROUP BY p.product_id, p.product_name
ORDER BY times_ordered DESC;
```



Showing rows 0 - 18 (19 total, Query took 0.0004 seconds.)

```
-- Query to aquire the list of products which have been ordered the most number of times -- (SQL Query)-- SELECT p.product_id, p.product_name, COUNT(oi.order_item_id) AS times_ordered FROM Products p JOIN Order_Items oi ON p.product_id = oi.product_id GROUP BY p.product_id, p.product_name ORDER BY times_ordered DESC;
```

product_id	product_name	times_ordered
18	Gaming Console	5
11	Mouse	3
1	Desktop	2
2	Headphones	2
3	Monitor	2
9	Book	2
12	External Hard Drive	2
13	USB Flash Drive	2
14	Smart TV	2
15	Portable Speaker	2
16	Charger	2
17	Laptop Bag	1
19	Video Game	1
20	Smartphone Case	1
4	Tablet	1
5	Camera	1
6	Watch	1
7	Backpack	1
8	Sunglasses	1
