

SQL ASSIGNMENT-1

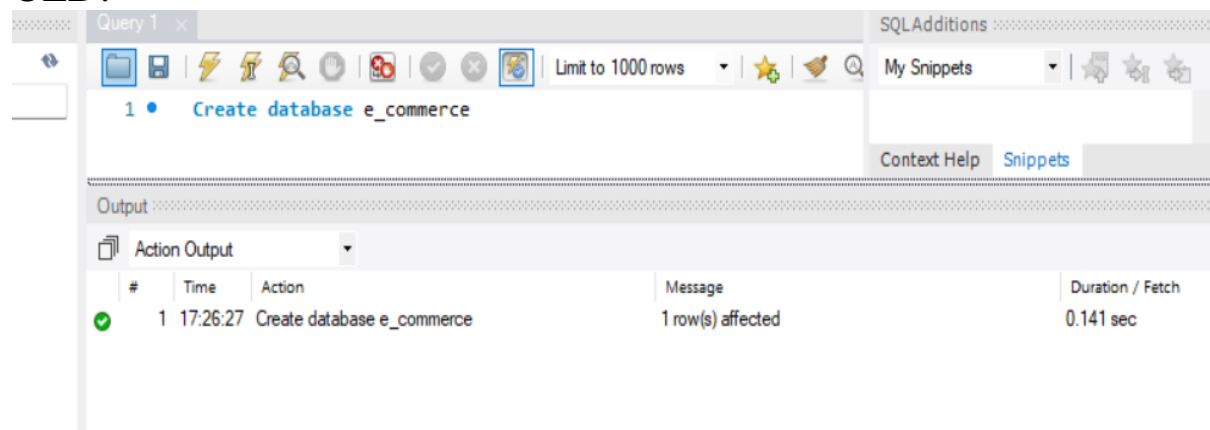
By: Akshat Saxena

1. Create Database e_commerce

NEW UPDATED:

```
CREATE DATABASE e_commerce; -- created database
```

OLD:



2. Create following Tables:

Customers:

- customer_id - int auto-increment primary key
- name - varchar(50)
- email - varchar(50)
- mobile - varchar(15)

```

3 • USE e_commerce;
4 • Create table Customers(
5   customer_id INT AUTO_INCREMENT PRIMARY KEY,
6   name VARCHAR(50) ,
7   email VARCHAR(50),
8   mobile VARCHAR(15)
9 );

```

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	17:26:27	Create database e_commerce	1 row(s) affected	0.141 sec
✗ 2	17:39:24	Create table Customers(customer_id INT AUTO_IN...	Error Code: 1046. No database selected Select the ...	0.062 sec
✓ 3	17:40:02	USE e_commerce	0 row(s) affected	0.109 sec
✓ 4	17:40:02	Create table Customers(customer_id INT AUTO_IN...	Create table Customers(customer_id INT AUTO_INCREMENT PRIMARY KEY, name VARCHAR(50) , email VARCHAR(50), mobile VARCHAR(15))	0.203 sec

Products:

- id - int
- name - varchar(50) not null
- description - varchar(200)
- price - decimal(10, 2) not null
- category - varchar(50)

```

10
11 • USE e_commerce;
12 • CREATE TABLE Products (
13   id INT,
14   name VARCHAR(50) NOT NULL,
15   description varchar(200),
16   price DECIMAL(10,2) NOT NULL,
17   category VARCHAR(50)
18 );
19

```

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 1	18:20:50	USE e_commerce	0 row(s) affected	0.063 sec
✓ 2	18:20:50	CREATE TABLE Products (id INT, name VAR...	0 row(s) affected	0.172 sec

3. Modify Tables(using Alter keyword):

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

```
ALTER TABLE Customers MODIFY name VARCHAR(50) NOT NULL;  
ALTER TABLE Customers MODIFY email VARCHAR(50) NOT NULL;
```

- b. Add unique key on email in the Customers table

```
ALTER TABLE Customers ADD CONSTRAINT unique_email UNIQUE(email);
```

- c. Add column age in the Customers table

```
ALTER TABLE Customers ADD COLUMN age INT;
```

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

```
ALTER TABLE Products CHANGE COLUMN id product_id INT;
```

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

```
ALTER TABLE  
Products  
MODIFY COLUMN product_id INT AUTO_INCREMENT PRIMARY KEY;
```

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

```
ALTER TABLE Products MODIFY description TEXT;
```

4. Create table Order:

- a. order_id - int auto-increment primary key
- b. customer_id - int -foreign key
- c. product_id - int
- d. quantity - int not null,

- e. order_date - date not null,
- f. status - enum(Pending, Success, Cancel),
- g. payment_method - enum(Credit, Debit, UPI),
- h. total_amount - decimal(10, 2) not null

The screenshot shows a SQL IDE with a script editor and an output window. The script editor contains the following SQL code:

```

29 • USE e_commerce;
30 • CREATE TABLE `Order` (
31     order_id INT AUTO_INCREMENT PRIMARY KEY,
32     customer_id INT,
33     product_id INT,
34     quantity INT NOT NULL,
35     order_date DATE NOT NULL,
36     status ENUM('Pending', 'Success', 'Cancel') DEFAULT 'Pending',
37     payment_method ENUM('Credit', 'Debit', 'UPI', 'COD'),
38     total_amount DECIMAL(10,2) NOT NULL,
39     FOREIGN KEY (customer_id) REFERENCES Customers(customer_id),
40     FOREIGN KEY (product_id) REFERENCES Products(product_id)
41 );
42
43

```

The output window shows a log of database actions:

#	Time	Action	Message	Duration / Fetch
7	18:25:13	ALTER TABLE Customers ADD CONSTRAINT unique_email U...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.032 sec
8	18:27:05	USE e_commerce	0 row(s) affected	0.000 sec
9	18:27:05	ALTER TABLE Products CHANGE id product_id INT AUTO_IN...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.047 sec
10	18:27:05	ALTER TABLE Products MODIFY description TEXT	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
11	18:36:45	USE e_commerce	0 row(s) affected	0.063 sec
12	18:36:45	CREATE TABLE `Order` (order_id INT AUTO_INCREMENT ...	0 row(s) affected	0.219 sec

5. Modify Orders Table(using Alter keyword):

- a. Change table name Order -> Orders

```
ALTER TABLE `Order` RENAME TO Orders;
```

- b. Set default value pending in status.

```
ALTER TABLE Orders
```

```
MODIFY status ENUM('Pending', 'Success', 'Cancel') DEFAULT 'Pending';
```

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

```
ALTER TABLE Orders
```

```
MODIFY payment_method ENUM('Credit', 'Debit', 'UPI', 'COD');
```

d. Make product id as foreign key

```
ALTER TABLE Orders
ADD CONSTRAINT fk_product FOREIGN KEY (product_id) REFERENCES Products(product_id);
```

✓	14 18:40:49	ALTER TABLE 'Order' RENAME TO Orders	0 row(s) affected
✓	15 18:40:49	ALTER TABLE Orders MODIFY status ENUM('Pending', 'Success', 'Cancel') DEFAULT 'Pending'	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓	16 18:40:49	ALTER TABLE Orders MODIFY payment_method ENUM('Credit', 'Debit', 'UPI', 'COD')	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓	17 18:40:49	ALTER TABLE Orders ADD CONSTRAINT fk_product FOREIGN KEY (product_id) REFERENCES Products(product_id)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

6. Insert 20 sample records in all the tables.

Data inserted in customers: (Initially I used query for insertion but didn't know that we have to paste the query here too as generally we have to show the output of the code. Sorry for the misunderstanding.)

```
INSERT INTO Customers (name, email, mobile, age) VALUES
('Alice Brown', 'alice@example.com', '1234567890', 30),
('Bob Smith', 'bob@example.com', '9876543210', 25),
('Charlie Davis', 'charlie@example.com', '4567891230', 35),
('David Wilson', 'david@example.com', '7891234560', 28),
('Emma Johnson', 'emma@example.com', '8529637410', 22),
('Frank Harris', 'frank@example.com', '9638527410', 40),
('Grace Lee', 'grace@example.com', '7418529630', 33),
('Henry White', 'henry@example.com', '1592634870', 29),
('Isabella Green', 'isabella@example.com', '7539518520', 26),
('Jack Taylor', 'jack@example.com', '8521479630', 37),
('Karen Miller', 'karen@example.com', '9517538520', 32),
('Leo King', 'leo@example.com', '7896541230', 27),
('Mia Scott', 'mia@example.com', '3214569870', 31),
('Nathan Young', 'nathan@example.com', '6547893210', 24),
('Olivia Adams', 'olivia@example.com', '1472583690', 36),
('Paul Nelson', 'paul@example.com', '8529631470', 29),
('Quinn Carter', 'quinn@example.com', '9637412580', 41),
('Rachel Brooks', 'rachel@example.com', '2581473690', 34),
('Sam Turner', 'sam@example.com', '1473692580', 30),
('Tina Evans', 'tina@example.com', '3692581470', 28);
```


Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	customer_id	name	email	mobile	age
▶	1	Alice Brown	alice@example.com	1234567890	30
	2	Bob Smith	bob@example.com	9876543210	25
	3	Charlie Davis	charlie@example.com	4567891230	35
	4	David Wilson	david@example.com	7891234560	28
	5	Emma Johnson	emma@example.com	8529637410	22
	6	Frank Harris	frank@example.com	9638527410	40
	7	Grace Lee	grace@example.com	7418529630	33
	8	Henry White	henry@example.com	1592634870	29
	9	Isabella Green	isabella@example.com	7539518520	26
	10	Jack Taylor	jack@example.com	8521479630	37
	11	Karen Miller	karen@example.com	9517536	9517538520
	12	Leo King	leo@example.com	7896541230	27
	13	Mia Scott	mia@example.com	3214569870	31
	14	Nathan Young	nathan@example.com	6547893210	24
	15	Olivia Adams	olivia@example.com	1472583690	36
	16	Paul Nelson	paul@example.com	8529631470	29
	17	Quinn Carter	quinn@example.com	9637412580	41
	18	Rachel Brooks	rachel@example.com	2581473690	34
	19	Sam Turner	sam@example.com	1473692580	30
	20	Tina Evans	tina@example.com	3692581470	28
*	NULL	NULL	NULL	NULL	NULL

Data inserted in products table:

- INSERT INTO** Products (name, description, price, category) **VALUES**
('Laptop', 'High performance laptop', 799.99, 'Electronics'),
('Smartphone', 'Latest model smartphone', 499.99, 'Electronics'),
('Headphones', 'Noise-canceling headphones', 199.99, 'Electronics'),
('Mouse', 'Wireless ergonomic mouse', 49.99, 'Electronics'),
('Keyboard', 'Mechanical gaming keyboard', 89.99, 'Electronics'),
('Tablet', '10-inch display tablet', 299.99, 'Electronics'),
('Monitor', '24-inch LED monitor', 150.00, 'Electronics'),
('Smartwatch', 'Fitness tracking smartwatch', 179.99, 'Electronics'),
('Printer', 'All-in-one wireless printer', 129.99, 'Electronics'),
('USB Drive', '128GB USB 3.0 drive', 25.99, 'Electronics'),
('Gaming Console', 'Next-gen gaming console', 499.99, 'Gaming'),
('VR Headset', 'Virtual reality headset', 299.99, 'Gaming'),
('Camera', 'DSLR camera', 699.99, 'Photography'),
('Tripod', 'Lightweight tripod', 89.99, 'Photography'),
('Backpack', 'Travel backpack', 59.99, 'Accessories'),
('Desk Lamp', 'LED desk lamp', 39.99, 'Home & Office'),
('Chair', 'Ergonomic office chair', 149.99, 'Furniture'),
('Sofa', 'Leather sofa', 999.99, 'Furniture'),
('Coffee Maker', 'Espresso coffee machine', 129.99, 'Appliances'),
('Blender', 'Multi-speed blender', 79.99, 'Appliances');

Result Grid					
Filter Rows:					
Edit: Export/Import: Wrap Cell Content:					
	product_id	name	description	price	category
▶	1	Laptop	High performance laptop	799.99	Electronics
	2	Smartphone	Latest model smartphone	499.99	Electronics
	3	Headphones	Noise-canceling headphones	199.99	Electronics
	4	Mouse	Wireless ergonomic mouse	49.99	Electronics
	5	Keyboard	Mechanical gaming keyboard	89.99	Electronics
	6	Tablet	10-inch display tablet	299.99	Electronics
	7	Monitor	24-inch LED monitor	150.00	Electronics
	8	Smartwatch	Fitness tracking smartwatch	179.99	Electronics
	9	Printer	All-in-one wireless printer	129.99	Electronics
	10	USB Drive	128GB USB 3.0 drive	25.99	Electronics
	11	Gaming Con...	Next-gen gaming console	499.99	Gaming
	12	VR Headset	Virtual reality headset	299.99	Gaming
	13	Camera	DSLR camera	699.99	Photogra...
	14	Tripod	Lightweight tripod	89.99	Photogra...
	15	Backpack	Travel backpack	59.99	Accessories
	16	Desk Lamp	LED desk lamp	39.99	Home & O...
	17	Chair	Ergonomic office chair	149.99	Furniture
	18	Sofa	Leather sofa	999.99	Furniture
	19	Coffee Maker	Espresso coffee machine	129.99	Appliances
	20	Blender	Multi-speed blender	79.99	Appliances
•	NOTE	NOTE	NOTE	NOTE	NOTE

Products 2 x

Data inserted in orders table:

```

3 • INSERT INTO Orders (customer_id, product_id, quantity, order_date, status, payment_method, total_amount) VALUES
3 (1, 1, 1, '2024-02-01', 'Success', 'Credit', 799.99),
3 (2, 2, 1, '2024-02-02', 'Pending', 'Debit', 499.99),
1 (3, 3, 2, '2024-02-03', 'Success', 'UPI', 399.98),
2 (4, 4, 1, '2024-02-04', 'Cancel', 'Credit', 49.99),
3 (5, 5, 1, '2024-02-05', 'Success', 'COD', 89.99),
4 (6, 6, 1, '2024-02-06', 'Success', 'UPI', 299.99),
5 (7, 7, 1, '2024-02-07', 'Pending', 'Credit', 150.00),
5 (8, 8, 2, '2024-02-08', 'Success', 'Debit', 359.98),
7 (9, 9, 1, '2024-02-09', 'Cancel', 'UPI', 129.99),
3 (10, 10, 3, '2024-02-10', 'Success', 'COD', 77.97),
3 (11, 11, 1, '2024-02-11', 'Success', 'Credit', 499.99),
3 (12, 12, 1, '2024-02-12', 'Pending', 'Debit', 299.99),
1 (13, 13, 2, '2024-02-13', 'Success', 'UPI', 1399.98),
2 (14, 14, 1, '2024-02-14', 'Cancel', 'Credit', 89.99),
3 (15, 15, 1, '2024-02-15', 'Success', 'COD', 59.99),
4 (16, 16, 1, '2024-02-16', 'Success', 'UPI', 39.99),
5 (17, 17, 1, '2024-02-17', 'Pending', 'Credit', 149.99),
5 (18, 18, 2, '2024-02-18', 'Success', 'Debit', 1999.98),
7 (19, 19, 1, '2024-02-19', 'Cancel', 'UPI', 129.99),
3 (20, 20, 3, '2024-02-20', 'Success', 'COD', 239.97);

```

Result Grid								
		Filter Rows:			Edit:		Export/Import:	
							Wrap Cell Content:	
	order_id	customer_id	product_id	quantity	order_date	status	payment_method	total_amount
▶	1	1	1	1	2024-02-01	Success	Credit	799.99
	2	2	2	1	2024-02-02	Pending	Debit	499.99
	3	3	3	2	2024-02-03	Success	UPI	399.98
	4	4	4	1	2024-02-04	Cancel	Credit	49.99
	5	5	5	1	2024-02-05	Success	COD	89.99
	6	6	6	1	2024-02-06	Success	UPI	299.99
	7	7	7	1	2024-02-07	Pending	Credit	150.00
	8	8	8	2	2024-02-08	Success	Debit	359.98
	9	9	9	1	2024-02-09	Cancel	UPI	129.99
	10	10	10	3	2024-02-10	Success	COD	77.97
	11	11	11	1	2024-02-11	Success	Credit	499.99
	12	12	12	1	2024-02-12	Pending	Debit	299.99
	13	13	13	2	2024-02-13	Success	UPI	1399.98
	14	14	14	1	2024-02-14	Cancel	Credit	89.99
	15	15	15	1	2024-02-15	Success	COD	59.99
	16	16	16	1	2024-02-16	Success	UPI	39.99
	17	17	17	1	2024-02-17	Pending	Credit	149.99
	18	18	18	2	2024-02-18	Success	Debit	1999.98
	19	19	19	1	2024-02-19	Cancel	UPI	129.99
	20	20	20	3	2024-02-20	Success	COD	239.97
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

7. Perform following queries:

- Count the number of products as product_count in each category.
(Query updated ac to standard.)

```

SELECT
    category, COUNT(*) AS product_count
FROM
    Products
GROUP BY category;

```


Result Grid			Filter Rows:	Export:
	category	product_count		
▶	Electronics	10		
	Gaming	2		
	Photography	2		
	Accessories	1		
	Home & Office	1		
	Furniture	2		
	Appliances	2		

b. Retrieve all products that belong to the 'Electronics' category, have a price between \$50 and \$500, and whose name contains the letter 'a'.

```

• SELECT
    *
FROM
    Products
WHERE
    category = 'Electronics'
    AND price BETWEEN 50 AND 500
    AND name LIKE '%a%';

```

(updated query above acc to basic coding standard.)

Result Grid					
Filter Rows:		Edit: Export/Import: Wrap Cell Center			
	product_id	name	description	price	category
▶	2	Smartphone	Latest model smartphone	499.99	Electronics
	3	Headphones	Noise-canceling headphones	199.99	Electronics
	5	Keyboard	Mechanical gaming keyboard	89.99	Electronics
	6	Tablet	10-inch display tablet	299.99	Electronics
	8	Smartwatch	Fitness tracking smartwatch	179.99	Electronics
*	NULL	NULL	NULL	NULL	NULL

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

```
SELECT
    *
FROM
    Products
WHERE
    category = 'Electronics'
ORDER BY price DESC
LIMIT 5 OFFSET 2;
```

(new updated code above acc to coding standard.)

Result Grid					
Filter Rows:		Edit: Export/Import:			
	product_id	name	description	price	category
▶	6	Tablet	10-inch display tablet	299.99	Electronics
	3	Headphones	Noise-canceling headphones	199.99	Electronics
	8	Smartwatch	Fitness tracking smartwatch	179.99	Electronics
	7	Monitor	24-inch LED monitor	150.00	Electronics
	9	Printer	All-in-one wireless printer	129.99	Electronics
*	NULL	NULL	NULL	NULL	NULL

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

```
SELECT  
    *  
  
FROM  
    Customers  
  
WHERE  
    customer_id NOT IN (SELECT DISTINCT  
                        customer_id  
                        FROM  
                            Orders);
```

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content

	customer_id	name	email	mobile	age
*	NULL	NULL	NULL	NULL	NULL

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

```
SELECT
    customer_id,
    AVG(total_amount) AS avg_spent
FROM
    Orders
GROUP BY customer_id;
```

187

Result Grid | Filter Rows: | Export:

	customer_id	avg_spent
▶	1	799.990000
	2	499.990000
	3	399.980000
	4	49.990000
	5	89.990000
	6	299.990000
	7	150.000000
	8	359.980000
	9	129.990000
	10	77.970000
	11	499.990000
	12	299.990000
	13	1399.980000
	14	89.990000
	15	59.990000
	16	39.990000
	17	149.990000
	18	1999.980000
	19	129.990000
	20	239.970000






Result 5 x

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

```

SELECT
    *
FROM
    Products
WHERE
    price < (SELECT
              AVG(price)
            FROM
              Products);

```

Result Grid					
Filter Rows: <input type="text"/>					
Edit:   					
Export/Import:  					
Wrap					
	product_id	name	description	price	category
▶	3	Headphones	Noise-canceling headphones	199.99	Electronics
	4	Mouse	Wireless ergonomic mouse	49.99	Electronics
	5	Keyboard	Mechanical gaming keyboard	89.99	Electronics
	7	Monitor	24-inch LED monitor	150.00	Electronics
	8	Smartwatch	Fitness tracking smartwatch	179.99	Electronics
	9	Printer	All-in-one wireless printer	129.99	Electronics
	10	USB Drive	128GB USB 3.0 drive	25.99	Electronics
	14	Tripod	Lightweight tripod	89.99	Photography
	15	Backpack	Travel backpack	59.99	Accessories
	16	Desk Lamp	LED desk lamp	39.99	Home & Office
	17	Chair	Ergonomic office chair	149.99	Furniture
	19	Coffee Maker	Espresso coffee machine	129.99	Appliances
	20	Blender	Multi-speed blender	79.99	Appliances
*	NULL	NULL	NULL	NULL	NULL

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

- ```

SELECT
 customer_id,
 SUM(quantity) AS total_quantity
FROM
 Orders
GROUP BY customer_id;

```



| Result Grid |             |                | Filter Rows: | Exp |
|-------------|-------------|----------------|--------------|-----|
|             | customer_id | total_quantity |              |     |
| ▶           | 1           | 1              |              |     |
|             | 2           | 1              |              |     |
|             | 3           | 2              |              |     |
|             | 4           | 1              |              |     |
|             | 5           | 1              |              |     |
|             | 6           | 1              |              |     |
|             | 7           | 1              |              |     |
|             | 8           | 2              |              |     |
|             | 9           | 1              |              |     |
|             | 10          | 3              |              |     |
|             | 11          | 1              |              |     |
|             | 12          | 1              |              |     |
|             | 13          | 2              |              |     |
|             | 14          | 1              |              |     |
|             | 15          | 1              |              |     |
|             | 16          | 1              |              |     |
|             | 17          | 1              |              |     |
|             | 18          | 2              |              |     |
|             | 19          | 1              |              |     |
|             | 20          | 3              |              |     |

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

- **SELECT**

```
o.order_id,
c.name AS customer_name,
p.name AS product_name,
o.quantity,
o.order_date,
o.status
```

**FROM**

Orders o

**JOIN**

Customers c

**ON** o.customer\_id = c.customer\_id

**JOIN**

Products p

**ON** o.product\_id = p.product\_id;

| Result Grid |          |                |                |          |            |                                       |
|-------------|----------|----------------|----------------|----------|------------|---------------------------------------|
|             |          | Filter Rows:   |                |          | Export:    | Wrap Cell Content: <a href="#">IA</a> |
|             | order_id | customer_name  | product_name   | quantity | order_date | status                                |
| ▶           | 1        | Alice Brown    | Laptop         | 1        | 2024-02-01 | Success                               |
|             | 2        | Bob Smith      | Smartphone     | 1        | 2024-02-02 | Pending                               |
|             | 3        | Charlie Davis  | Headphones     | 2        | 2024-02-03 | Success                               |
|             | 4        | David Wilson   | Mouse          | 1        | 2024-02-04 | Cancel                                |
|             | 5        | Emma Johnson   | Keyboard       | 1        | 2024-02-05 | Success                               |
|             | 6        | Frank Harris   | Tablet         | 1        | 2024-02-06 | Success                               |
|             | 7        | Grace Lee      | Monitor        | 1        | 2024-02-07 | Pending                               |
|             | 8        | Henry White    | Smartwatch     | 2        | 2024-02-08 | Success                               |
|             | 9        | Isabella Green | Printer        | 1        | 2024-02-09 | Cancel                                |
|             | 10       | Jack Taylor    | USB Drive      | 3        | 2024-02-10 | Success                               |
|             | 11       | Karen Miller   | Gaming Console | 1        | 2024-02-11 | Success                               |
|             | 12       | Leo King       | VR Headset     | 1        | 2024-02-12 | Pending                               |
|             | 13       | Mia Scott      | Camera         | 2        | 2024-02-13 | Success                               |
|             | 14       | Nathan Young   | Tripod         | 1        | 2024-02-14 | Cancel                                |
|             | 15       | Olivia Adams   | Backpack       | 1        | 2024-02-15 | Success                               |
|             | 16       | Paul Nelson    | Desk Lamp      | 1        | 2024-02-16 | Success                               |
|             | 17       | Quinn Carter   | Chair          | 1        | 2024-02-17 | Pending                               |
|             | 18       | Rachel Brooks  | Sofa           | 2        | 2024-02-18 | Success                               |
|             | 19       | Sam Turner     | Coffee Maker   | 1        | 2024-02-19 | Cancel                                |
|             | 20       | Tina Evans     | Blender        | 3        | 2024-02-20 | Success                               |

- i. Find products that have never been ordered.

```
SELECT
 *
FROM
 Products
WHERE
 product_id NOT IN (SELECT DISTINCT
 product_id
 FROM
 Orders);
```

|             |            |              |             |       |          |                |  |                    |  |
|-------------|------------|--------------|-------------|-------|----------|----------------|--|--------------------|--|
| Result Grid |            | Filter Rows: |             | Edit: |          | Export/Import: |  | Wrap Cell Content: |  |
|             | product_id | name         | description | price | category |                |  |                    |  |
| *           | NULL       | NULL         | NULL        | NULL  | NULL     |                |  |                    |  |

E\_commerce (EER\_Diagram for reference):

