

Coding challenge_6(Ecommerce)

1. Update refrigerator product price to 800.

Sol:-

```
mysql> select * from products;
```

product_id	name	price	description	stockquantity
1	laptop	900.00	high-performance laptop	10
2	smartphone	600.00	latest 5G andriod phone	20
3	tablet	500.00	portable and easy to handle	15
4	headphones	400.00	boat is brand now	30
5	TV	1000.00	4k smart tv with android	20
6	coffeeMaker	50.00	automatic coffee maker	23
7	refrigerator	700.00	Energy-efficient	10
8	microwave oven	90.00	countertop microwave	15
9	blender	70.00	high-speed blender	20
10	vacuum cleaner	120.00	bagless vacuum cleaner	10

```
mysql> update products
-> SET price=800
-> where name='refrigerator';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> select * from products;
```

product_id	name	price	description	stockquantity
1	laptop	900.00	high-performance laptop	10
2	smartphone	600.00	latest 5G andriod phone	20
3	tablet	500.00	portable and easy to handle	15
4	headphones	400.00	boat is brand now	30
5	TV	1000.00	4k smart tv with android	20
6	coffeeMaker	50.00	automatic coffee maker	23
7	refrigerator	800.00	Energy-efficient	10
8	microwave oven	90.00	countertop microwave	15
9	blender	70.00	high-speed blender	20
10	vacuum cleaner	120.00	bagless vacuum cleaner	10

10 rows in set (0.00 sec)

2. Remove all cart items for a specific customer.

Sol:-

```
mysql> select * from cart;
```

cart_id	customer_id	product_id	quantity
1	1	1	2
2	1	3	1
3	2	2	3
4	2	4	2
5	3	5	1
6	3	7	4
7	4	6	2
8	4	8	3
9	5	9	1
10	5	10	2

10 rows in set (0.00 sec)

```
mysql> delete from cart
      -> where customer_id=2;
Query OK, 2 rows affected (0.01 sec)

mysql> select * from cart;
+-----+-----+-----+-----+
| cart_id | customer_id | product_id | quantity |
+-----+-----+-----+-----+
| 1 | 1 | 1 | 2 |
| 2 | 1 | 3 | 1 |
| 5 | 3 | 5 | 1 |
| 6 | 3 | 7 | 4 |
| 7 | 4 | 6 | 2 |
| 8 | 4 | 8 | 3 |
| 9 | 5 | 9 | 1 |
| 10 | 5 | 10 | 2 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

3. Retrieve Products Priced Below \$100.

Sol:-

```
mysql> select * from products;
+-----+-----+-----+-----+-----+
| product_id | name | price | description | stockquantity |
+-----+-----+-----+-----+-----+
| 1 | laptop | 900.00 | high-performance laptop | 10 |
| 2 | smartphone | 600.00 | latest 5G andriod phone | 20 |
| 3 | tablet | 500.00 | portable and easy to handle | 15 |
| 4 | headphones | 400.00 | boat is brand now | 30 |
| 5 | TV | 1000.00 | 4k smart tv with android | 20 |
| 6 | coffeeMaker | 50.00 | automatic coffee maker | 23 |
| 7 | refrigerator | 800.00 | Energy-efficient | 10 |
| 8 | microwave oven | 90.00 | countertop microwave | 15 |
| 9 | blender | 70.00 | high-speed blender | 20 |
| 10 | vacuum cleaner | 120.00 | bagless vacuum cleaner | 10 |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql> select * from products
      -> where price<100;
+-----+-----+-----+-----+-----+
| product_id | name | price | description | stockquantity |
+-----+-----+-----+-----+-----+
| 6 | coffeeMaker | 50.00 | automatic coffee maker | 23 |
| 8 | microwave oven | 90.00 | countertop microwave | 15 |
| 9 | blender | 70.00 | high-speed blender | 20 |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

4. Find Products with Stock Quantity Greater Than 5.

Sol:- I am taking stock quantity grater than 10 because my all products stock quantity is greater than 5..

```
mysql> select * from products;
```

product_id	name	price	description	stockquantity
1	laptop	900.00	high-performance laptop	10
2	smartphone	600.00	latest 5G andriod phone	20
3	tablet	500.00	portable and easy to handle	15
4	headphones	400.00	boat is brand now	30
5	TV	1000.00	4k smart tv with android	20
6	coffeeMaker	50.00	automatic coffee maker	23
7	refrigerator	800.00	Energy-efficient	10
8	microwave oven	90.00	countertop microwave	15
9	blender	70.00	high-speed blender	20
10	vacuum cleaner	120.00	bagless vacuum cleaner	10

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

```
mysql> select * from products
-> where stockquantity>10;
```

product_id	name	price	description	stockquantity
2	smartphone	600.00	latest 5G andriod phone	20
3	tablet	500.00	portable and easy to handle	15
4	headphones	400.00	boat is brand now	30
5	TV	1000.00	4k smart tv with android	20
6	coffeeMaker	50.00	automatic coffee maker	23
8	microwave oven	90.00	countertop microwave	15
9	blender	70.00	high-speed blender	20

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

5. Retrieve Orders with Total Amount Between \$500 and \$1000.

Sol:-

```
mysql> select * from orders;
```

order_id	customer_id	order_date	total_price	shipping_address
1	1	2024-01-23	1200.00	Sohjana near block
2	2	2024-01-24	900.00	karpuri chowk jhajha
3	3	2024-01-25	1100.00	old bus stand near CYX
4	4	2024-01-26	800.00	khalasi mohalla
5	5	2024-01-27	1500.00	station club station road
6	6	2024-01-28	799.00	sis mahal main bazar
7	7	2024-01-29	400.00	balaji store
8	8	2024-01-30	550.00	pihar mor
9	9	2024-01-31	1400.00	durga mandir jhajha
10	10	2024-02-01	999.00	gandhi chowk jhajha

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

Coding challenge_6(Ecommerce)

```
mysql> select * from orders
-> where total_price BETWEEN 500 AND 1000;
+-----+-----+-----+-----+-----+
| order_id | customer_id | order_date | total_price | shipping_address |
+-----+-----+-----+-----+-----+
| 2 | 2 | 2024-01-24 | 900.00 | karpuri chowk jhajha |
| 4 | 4 | 2024-01-26 | 800.00 | khalasi mohalla |
| 6 | 6 | 2024-01-28 | 799.00 | sis mahal main bazar |
| 8 | 8 | 2024-01-30 | 550.00 | pihar mor |
| 10 | 10 | 2024-02-01 | 999.00 | gandhi chowk jhajha |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

6. Find Products which name end with letter 'r'.

Sol:-

```
mysql> select * from products;
+-----+-----+-----+-----+-----+
| product_id | name | price | description | stockquantity |
+-----+-----+-----+-----+-----+
| 1 | laptop | 900.00 | high-performance laptop | 10 |
| 2 | smartphone | 600.00 | latest 5G andriod phone | 20 |
| 3 | tablet | 500.00 | portable and easy to handle | 15 |
| 4 | headphones | 400.00 | boat is brand now | 30 |
| 5 | TV | 1000.00 | 4k smart tv with android | 20 |
| 6 | coffeeMaker | 50.00 | automatic coffee maker | 23 |
| 7 | refrigerator | 800.00 | Energy-efficient | 10 |
| 8 | microwave oven | 90.00 | countertop microwave | 15 |
| 9 | blender | 70.00 | high-speed blender | 20 |
| 10 | vacuum cleaner | 120.00 | bagless vacuum cleaner | 10 |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql> select * from products
-> where name like '%r';
+-----+-----+-----+-----+-----+
| product_id | name | price | description | stockquantity |
+-----+-----+-----+-----+-----+
| 6 | coffeeMaker | 50.00 | automatic coffee maker | 23 |
| 7 | refrigerator | 800.00 | Energy-efficient | 10 |
| 9 | blender | 70.00 | high-speed blender | 20 |
| 10 | vacuum cleaner | 120.00 | bagless vacuum cleaner | 10 |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

7. Retrieve Cart Items for Customer 5.

Sol:-

```
mysql> select * from cart;
+-----+-----+-----+-----+
| cart_id | customer_id | product_id | quantity |
+-----+-----+-----+-----+
| 1 | 1 | 1 | 2 |
| 2 | 1 | 3 | 1 |
| 5 | 3 | 5 | 1 |
| 6 | 3 | 7 | 4 |
| 7 | 4 | 6 | 2 |
| 8 | 4 | 8 | 3 |
| 9 | 5 | 9 | 1 |
| 10 | 5 | 10 | 2 |
+-----+-----+-----+-----+
8 rows in set (0.00 sec)
```

```
mysql> select * from cart
-> where customer_id=5;
```

cart_id	customer_id	product_id	quantity
9	5	9	1
10	5	10	2

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

8. Find Customers Who Placed Orders in 2023

Sol:-

```
mysql> select * from orders;
```

order_id	customer_id	order_date	total_price	shipping_address
1	1	2024-01-23	1200.00	Sohjana near block
2	2	2024-01-24	900.00	karpuri chowk jhajha
3	3	2024-01-25	1100.00	old bus stand near CYX
4	4	2024-01-26	800.00	khalasi mohalla
5	5	2024-01-27	1500.00	station club station road
6	6	2024-01-28	799.00	sis mahal main bazar
7	7	2024-01-29	400.00	balaji store
8	8	2024-01-30	550.00	pihar mor
9	9	2024-01-31	1400.00	durga mandir jhajha
10	10	2024-02-01	999.00	gandhi chowk jhajha

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

```
mysql> select distinct c.* from customers c
-> join orders o On c.customer_id=o.customer_id
-> where YEAR(o.order_date)=2023;
Empty set (0.00 sec)

mysql> select distinct c.* from customers c
-> join orders o On c.customer_id=o.customer_id
-> where YEAR(o.order_date)=2024;
```

customer_id	name	email	password
1	sourav	srv@gmail.com	sourav123
2	rishav	ris@gmail.com	rishav123
3	subham	sub@gmail.com	subham123
4	mayank	mayank@gmail.com	mayank123
5	stayam	sat@gmail.com	satyam123
6	rahul	rah@gmail.com	rahul123
7	abhishek	jha@gmail.com	abhishek123
8	rocky	rok@gmail.com	rocky123
9	amit	ami@gmail.com	amit123
10	purusottam	puru@gmail.com	purusottam123

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

9. Determine the Minimum Stock Quantity for Each Product Category.**Sol:-**

```
mysql> create table categories(  
  -> category_id INT PRIMARY KEY,  
  -> category_name varchar(50) UNIQUE  
  -> );  
Query OK, 0 rows affected (0.04 sec)  
  
mysql> alter table products  
  -> add column category_id INT,  
  -> add CONSTRAINT fk_category  
  -> foreign key(category_id) references categories(category_id);  
Query OK, 10 rows affected (0.07 sec)  
Records: 10  Duplicates: 0  Warnings: 0
```

```
mysql> insert into categories values  
  -> (1,'electronics'),  
  -> (2,'garden'),  
  -> (3,'clothing'),  
  -> (4,'spots'),  
  -> (5,'toys'),  
  -> (6,'books'),  
  -> (7,'auto'),  
  -> (8,'furniture'),  
  -> (9,'gym'),  
  -> (10,'home');  
Query OK, 10 rows affected (0.00 sec)  
Records: 10  Duplicates: 0  Warnings: 0
```

```
mysql> update products  
  -> SET category_id= CASE  
  -> WHEN product_id=1 THEN 1  
  -> WHEN product_id=2 THEN 1  
  -> WHEN product_id=3 THEN 1  
  -> WHEN product_id=4 THEN 1  
  -> WHEN product_id=5 THEN 1  
  -> WHEN product_id=6 THEN 1  
  -> WHEN product_id=7 THEN 1  
  -> WHEN product_id=8 THEN 1  
  -> WHEN product_id=9 THEN 1  
  -> WHEN product_id=10 THEN 1  
  -> END;  
Query OK, 10 rows affected (0.00 sec)  
Rows matched: 10  Changed: 10  Warnings: 0
```

Coding challenge_6(Ecommerce)

```
mysql> select c.category_id,c.category_name,MIN(p.stockquantity) As min_stock_quantity
-> from categories c
-> join products p ON c.category_id=p.category_id
-> group by c.category_id,c.category_name;
+-----+-----+-----+
| category_id | category_name | min_stock_quantity |
+-----+-----+-----+
|          1 | electronics   |          10        |
+-----+-----+-----+
1 row in set (0.00 sec)
```

10. Calculate the Total Amount Spent by Each Customer.

Sol:-

```
mysql> select customers.customer_id,customers.name,SUM(orders.total_price)AS
total_amount_spend
-> from orders
-> join customers ON orders.customer_id=customers.customer_id
-> group by customers.customer_id,customers.name;
+-----+-----+-----+
| customer_id | name          | AStotal_amount_spend |
+-----+-----+-----+
|          1 | sourav        |          1200.00      |
|          2 | rishav        |           900.00      |
|          3 | subham        |          1100.00      |
|          4 | mayank        |           800.00      |
|          5 | stayam        |          1500.00      |
|          6 | rahul         |           799.00      |
|          7 | abhishek      |           400.00      |
|          8 | rocky         |           550.00      |
|          9 | amit          |          1400.00      |
|         10 | purusottam    |           999.00      |
+-----+-----+-----+
10 rows in set (0.00 sec)
```

11. Find the Average Order Amount for Each Customer.

Sol:-

```
mysql> select customers.customer_id,customers.name,AVG(orders.total_price)AS
average_order_amount from orders
-> join customers ON orders.customer_id=customers.customer_id
-> group by customers.customer_id,customers.name;
+-----+-----+-----+
| customer_id | name          | ASaverage_order_amount |
+-----+-----+-----+
|          1 | sourav        |          1200.000000   |
|          2 | rishav        |           900.000000   |
|          3 | subham        |          1100.000000   |
|          4 | mayank        |           800.000000   |
|          5 | stayam        |          1500.000000   |
|          6 | rahul         |           799.000000   |
|          7 | abhishek      |           400.000000   |
|          8 | rocky         |           550.000000   |
|          9 | amit          |          1400.000000   |
|         10 | purusottam    |           999.000000   |
+-----+-----+-----+
10 rows in set (0.00 sec)
```

12. Count the Number of Orders Placed by Each Customer**Sol:-**

```
mysql> select customers.customer_id,customers.name,COUNT(orders.order_id)AS
number_of_orders from orders
-> join customers ON orders.customer_id=customers.customer_id
-> group by customers.customer_id,customers.name;
```

customer_id	name	number_of_orders
1	sourav	1
2	rishav	1
3	subham	1
4	mayank	1
5	stayam	1
6	rahul	1
7	abhishek	1
8	rocky	1
9	amit	1
10	purusottam	1

10 rows in set (0.00 sec)

13. Find the Maximum Order Amount for Each Customer.**Sol:-**

```
mysql> select customers.customer_id,customers.name,MAX(orders.total_price)AS
max_order_amount from orders
-> join customers ON orders.customer_id=customers.customer_id
-> group by customers.customer_id,customers.name;
```

customer_id	name	max_order_amount
1	sourav	1200.00
2	rishav	900.00
3	subham	1100.00
4	mayank	800.00
5	stayam	1500.00
6	rahul	799.00
7	abhishek	400.00
8	rocky	550.00
9	amit	1400.00
10	purusottam	999.00

10 rows in set (0.00 sec)

14. Get Customers Who Placed Orders Totaling Over \$1000.**Sol:-**

Coding challenge_6(Ecommerce)

```
mysql> select customers.customer_id,customers.name,SUM(orders.total_price)AS
total_amount_spend from orders
-> join customers ON orders.customer_id=customers.customer_id
-> group by customers.customer_id,customers.name
-> having total_amount_spend>1000;
```

customer_id	name	total_amount_spend
1	sourav	1200.00
3	subham	1100.00
5	stayam	1500.00
9	amit	1400.00

4 rows in set (0.00 sec)

15. Subquery to Find Products Not in the Cart.

Sol:-

```
mysql> select * from products;
```

product_id	name	price	description	stockquantity	category_id
1	laptop	900.00	high-performance laptop	10	1
2	smartphone	600.00	latest 5G andriod phone	20	1
3	tablet	500.00	portable and easy to handle	15	1
4	headphones	400.00	boat is brand now	30	1
5	TV	1000.00	4k smart tv with android	20	1
6	coffeeMaker	50.00	automatic coffee maker	23	1
7	refrigerator	800.00	Energy-efficient	10	1
8	microwave oven	90.00	countertop microwave	15	1
9	blender	70.00	high-speed blender	20	1
10	vacuum cleaner	120.00	bagless vacuum cleaner	10	1

```
mysql> select * from cart;
```

cart_id	customer_id	product_id	quantity
1	1	1	2
2	1	3	1
5	3	5	1
6	3	7	4
7	4	6	2
8	4	8	3
9	5	9	1
10	5	10	2

8 rows in set (0.00 sec)

```
mysql> select * from products
-> where product_id NOT IN(select product_id from cart);
```

product_id	name	price	description	stockquantity	category_id
2	smartphone	600.00	latest 5G andriod phone	20	1
4	headphones	400.00	boat is brand now	30	1

2 rows in set (0.01 sec)

16. Subquery to Find Customers Who Haven't Placed Orders

Sol:-

```
mysql> select * from customers;
```

customer_id	name	email	password
1	sourav	srv@gmail.com	sourav123
2	rishav	ris@gmail.com	rishav123
3	subham	sub@gmail.com	subham123
4	mayank	mayank@gmail.com	mayank123
5	stayam	sat@gmail.com	satyam123
6	rahul	rah@gmail.com	rahul123
7	abhishek	jha@gmail.com	abhishek123
8	rocky	rok@gmail.com	rocky123
9	amit	ami@gmail.com	amit123
10	purusottam	puru@gmail.com	purusottam123

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

```
mysql> select * from orders;
```

order_id	customer_id	order_date	total_price	shipping_address
1	1	2024-01-23	1200.00	Sohjana near block
2	2	2024-01-24	900.00	karpuri chowk jhajha
3	3	2024-01-25	1100.00	old bus stand near CYX
4	4	2024-01-26	800.00	khalasi mohalla
5	5	2024-01-27	1500.00	station club station road
6	6	2024-01-28	799.00	sis mahal main bazar
7	7	2024-01-29	400.00	balaji store
8	8	2024-01-30	550.00	pihar mor
9	9	2024-01-31	1400.00	durga mandir jhajha
10	10	2024-02-01	999.00	gandhi chowk jhajha

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

```
mysql> select * from customers
    -> where customer_id NOT IN(select customer_id from orders);
Empty set (0.00 sec)
```

17. Subquery to Calculate the Percentage of Total Revenue for a Product.

Sol:-percentage=(amount/total amount*100)

product_id	name	price	description	stockquantity	category_id
1	laptop	900.00	high-performance laptop	10	1
2	smartphone	600.00	latest 5G andriod phone	20	1
3	tablet	500.00	portable and easy to handle	15	1
4	headphones	400.00	boat is brand now	30	1
5	TV	1000.00	4k smart tv with android	20	1
6	coffeeMaker	50.00	automatic coffee maker	23	1
7	refrigerator	800.00	Energy-efficient	10	1
8	microwave oven	90.00	countertop microwave	15	1
9	blender	70.00	high-speed blender	20	1
10	vacuum cleaner	120.00	bagless vacuum cleaner	10	1

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

Coding challenge_6(Ecommerce)

```
mysql> select * from orders;
```

order_id	customer_id	order_date	total_price	shipping_address
1	1	2024-01-23	1200.00	Sohjana near block
2	2	2024-01-24	900.00	karpuri chowk jhajha
3	3	2024-01-25	1100.00	old bus stand near CYX
4	4	2024-01-26	800.00	khalasi mohalla
5	5	2024-01-27	1500.00	station club station road
6	6	2024-01-28	799.00	sis mahal main bazar
7	7	2024-01-29	400.00	balaji store
8	8	2024-01-30	550.00	pihar mor
9	9	2024-01-31	1400.00	durga mandir jhajha
10	10	2024-02-01	999.00	gandhi chowk jhajha

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

```
mysql> select product_id,name,price,(price/(select SUM(total_price) from orders)*100) AS percentage_total_revenue from products
-> where product_id=1;
```

product_id	name	price	percentage_total_revenue
1	laptop	900.00	9.328358

```
1 row in set (0.00 sec)
```

18. Subquery to Find Products with Low Stock.

Sol:-here, low stocks means less then the average stock quantity

```
mysql> select * from products
-> where stockquantity <(select AVG(stockquantity) from products);
```

product_id	name	price	description	stockquantity	category_id
1	laptop	900.00	high-performance laptop	10	1
3	tablet	500.00	portable and easy to handle	15	1
7	refrigerator	800.00	Energy-efficient	10	1
8	microwave oven	90.00	countertop microwave	15	1
10	vacuum cleaner	120.00	bagless vacuum cleaner	10	1

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

19. Subquery to Find Customers Who Placed High-Value Orders

Sol:-

```
mysql> select * from customers;
```

customer_id	name	email	password
1	sourav	srv@gmail.com	sourav123
2	rishav	ris@gmail.com	rishav123
3	subham	sub@gmail.com	subham123
4	mayank	mayank@gmail.com	mayank123
5	stayam	sat@gmail.com	satyam123
6	rahul	rah@gmail.com	rahul123
7	abhishek	jha@gmail.com	abhishek123
8	rocky	rok@gmail.com	rocky123
9	amit	ami@gmail.com	amit123
10	purusottam	puru@gmail.com	purusottam123

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

Coding challenge_6(Ecommerce)

```
mysql> select * from orders;
```

order_id	customer_id	order_date	total_price	shipping_address
1	1	2024-01-23	1200.00	Sohjana near block
2	2	2024-01-24	900.00	karpuri chowk jhajha
3	3	2024-01-25	1100.00	old bus stand near CYX
4	4	2024-01-26	800.00	khalasi mohalla
5	5	2024-01-27	1500.00	station club station road
6	6	2024-01-28	799.00	sis mahal main bazar
7	7	2024-01-29	400.00	balaji store
8	8	2024-01-30	550.00	pihar mor
9	9	2024-01-31	1400.00	durga mandir jhajha
10	10	2024-02-01	999.00	gandhi chowk jhajha

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

```
mysql> select * from customers
```

```
-> where customer_id IN(select customer_id from orders where total_price>1000);
```

customer_id	name	email	password
1	sourav	srv@gmail.com	sourav123
3	subham	sub@gmail.com	subham123
5	stayam	sat@gmail.com	satyam123
9	amit	ami@gmail.com	amit123

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