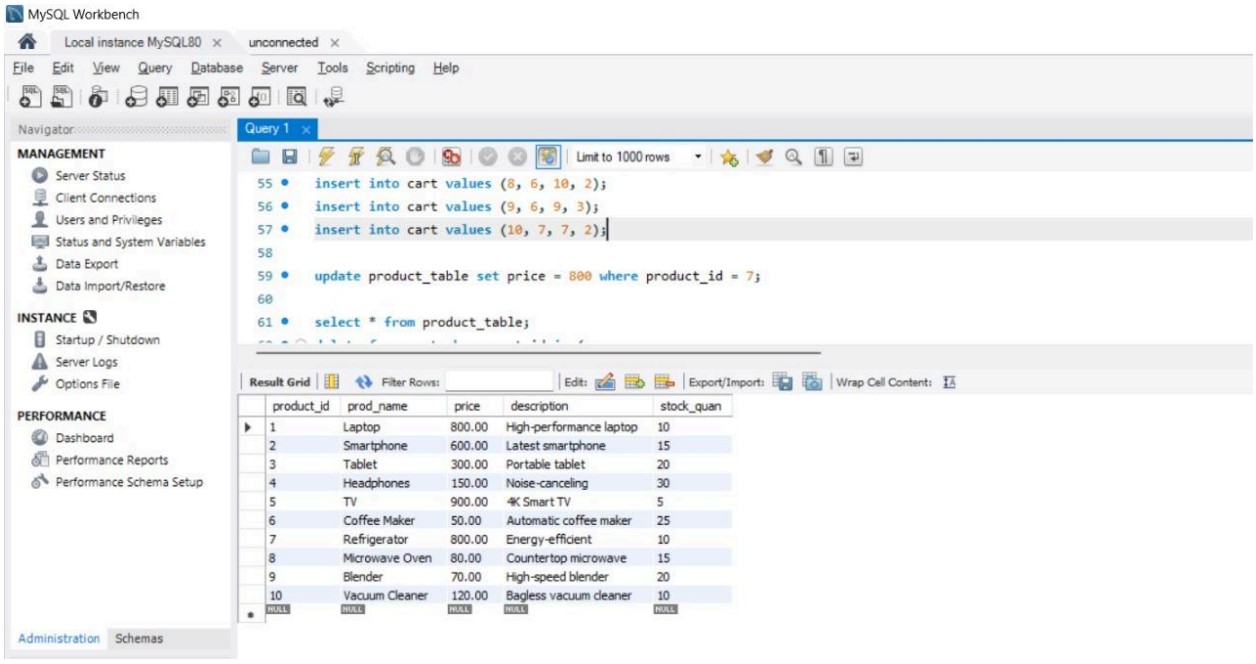
Ecommerce – SQL

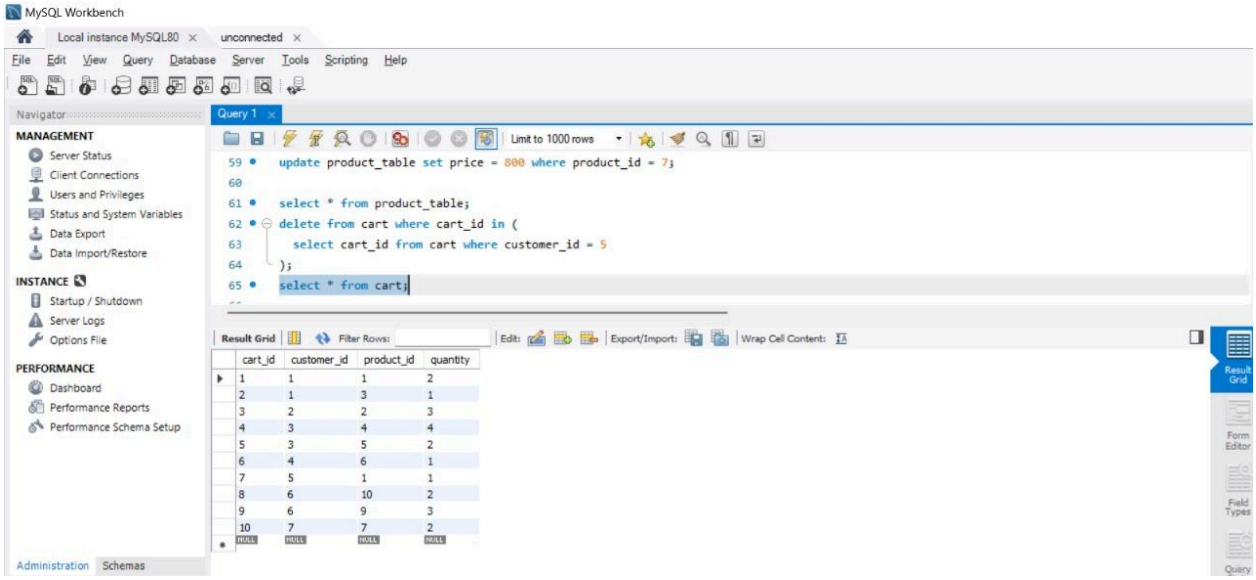
1.

1) update product_table set price = 800 where product_id = 7;



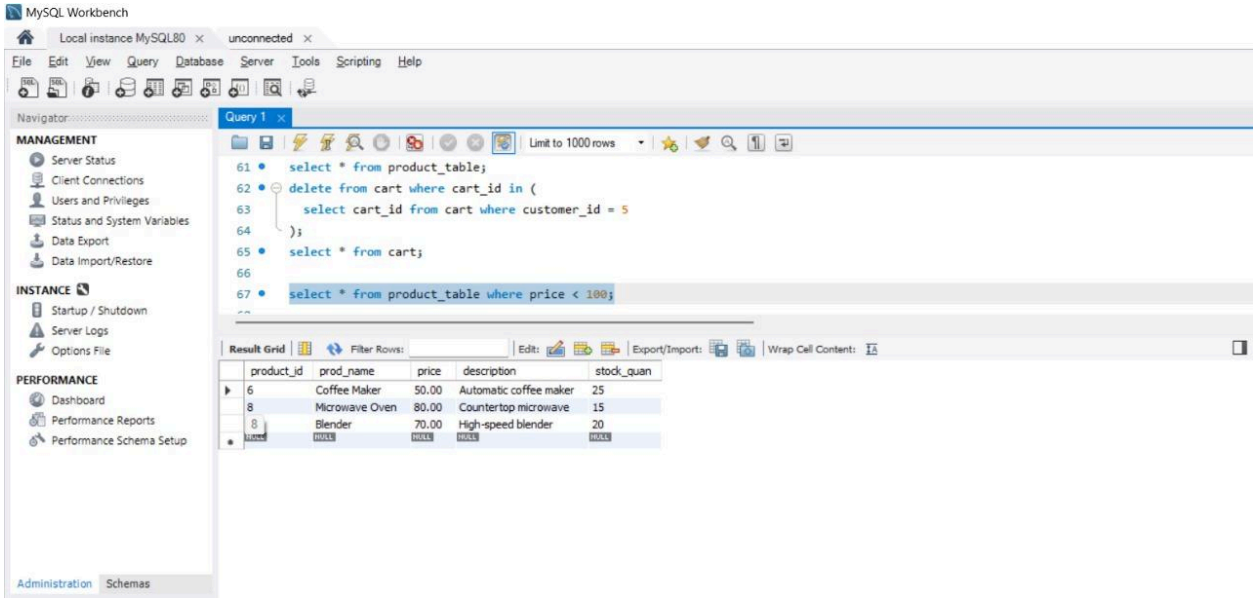
2.

2) delete from cart where cart_id in (select cart_id from cart where customer_id = 5);



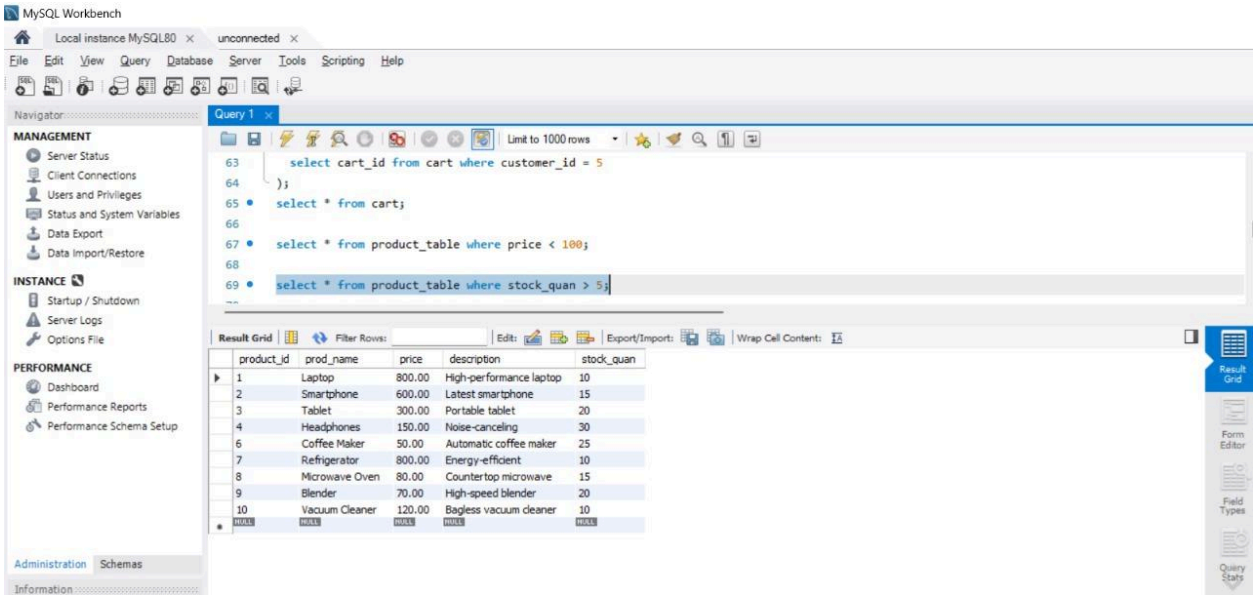
3.

3) select * from product_table where price < 100;

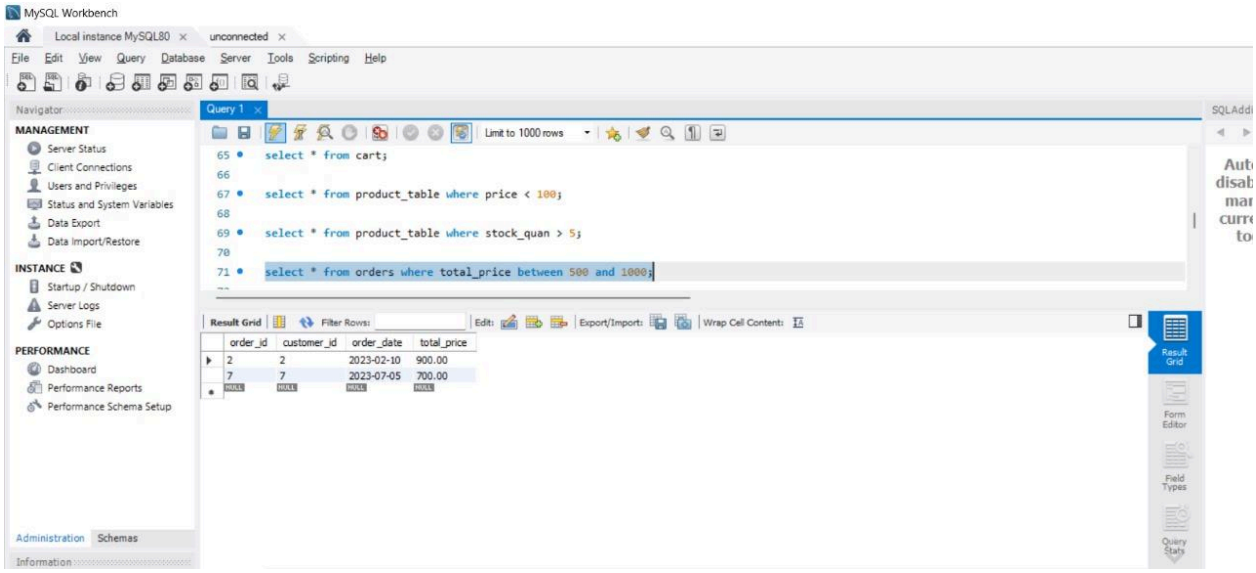


4.

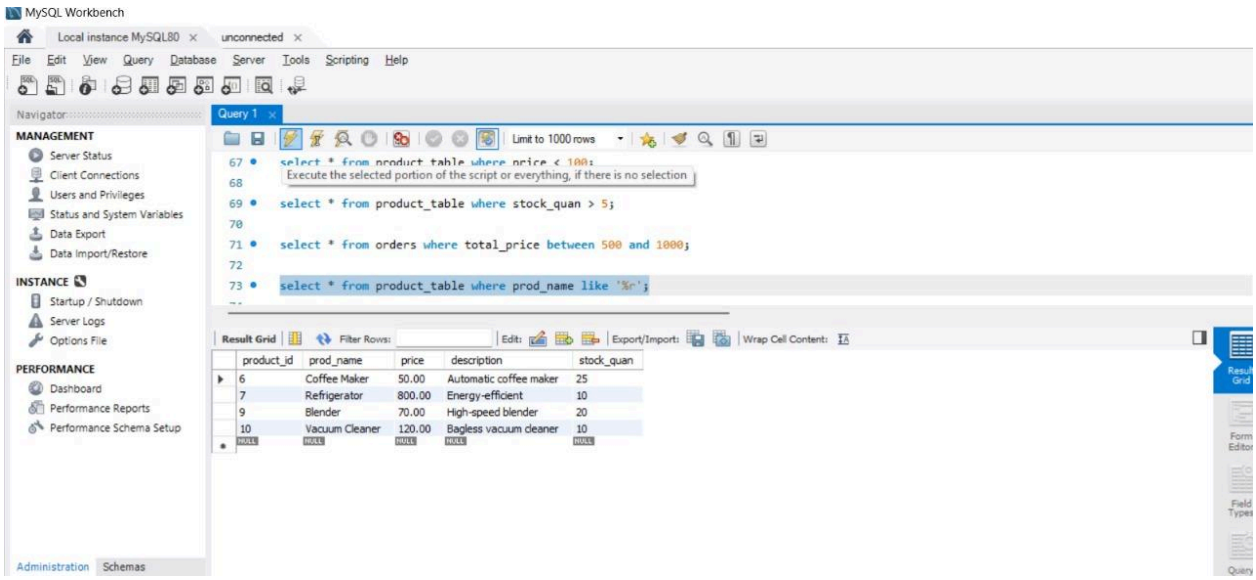
4)select * from product_table where stock_quan > 5;



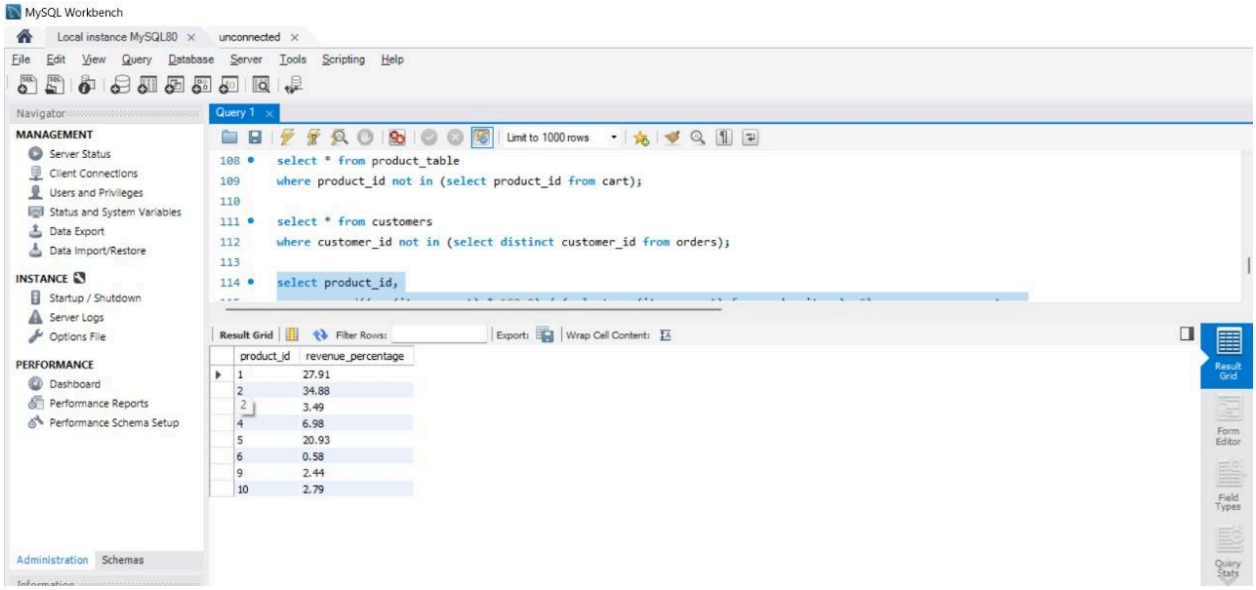
5) select * from orders where total_price between 500 and 1000;



6) select * from product_table where prod_name like '%r';

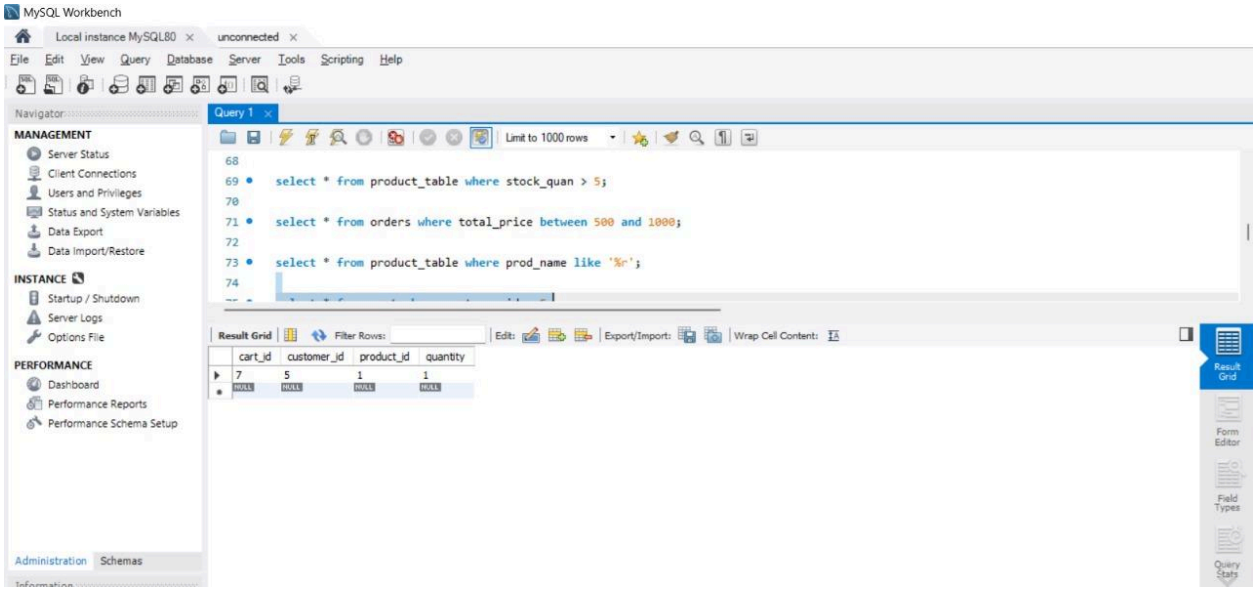


7) select * from cart where customer_id = 5;

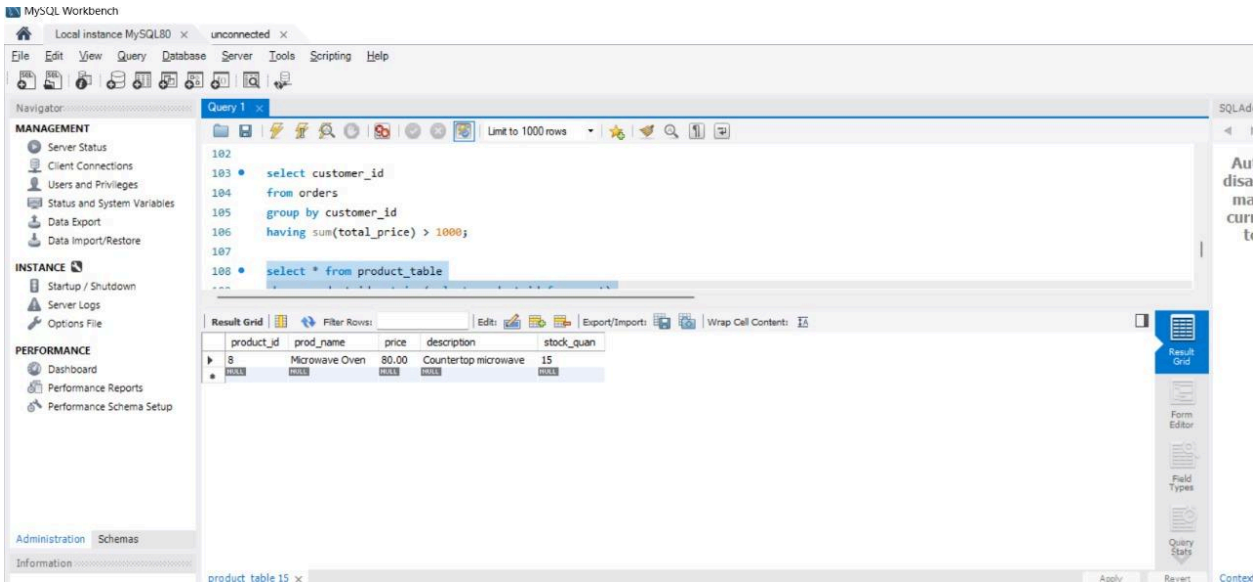


8.

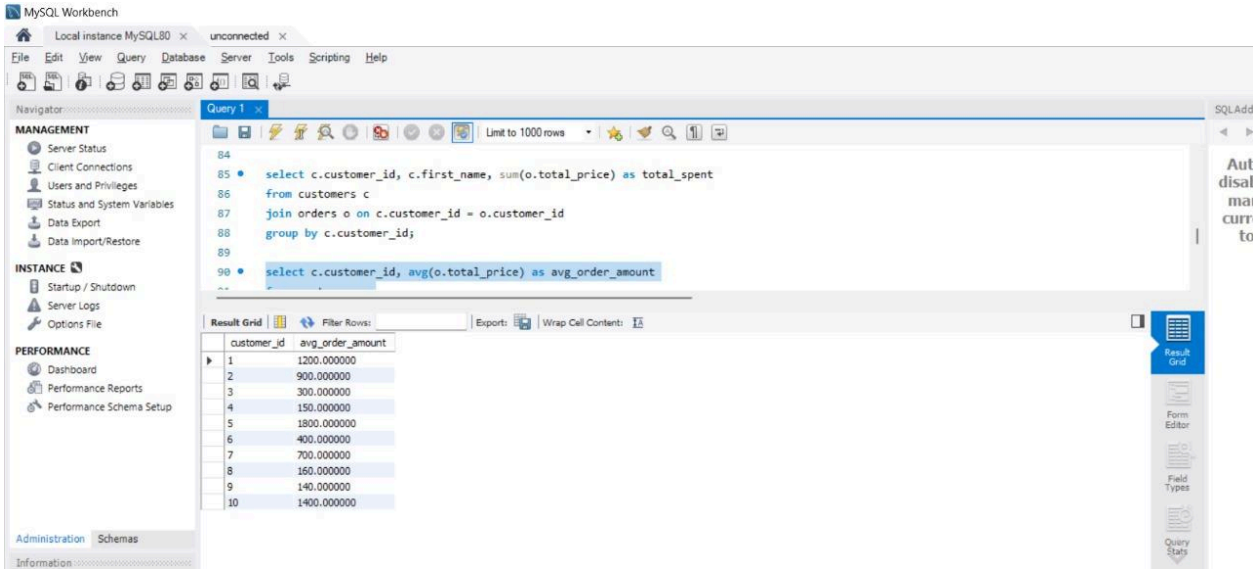
8) select distinct c.* from customers c join orders o on c.customer_id = o.customer_id where year(o.order_date) = 2023;



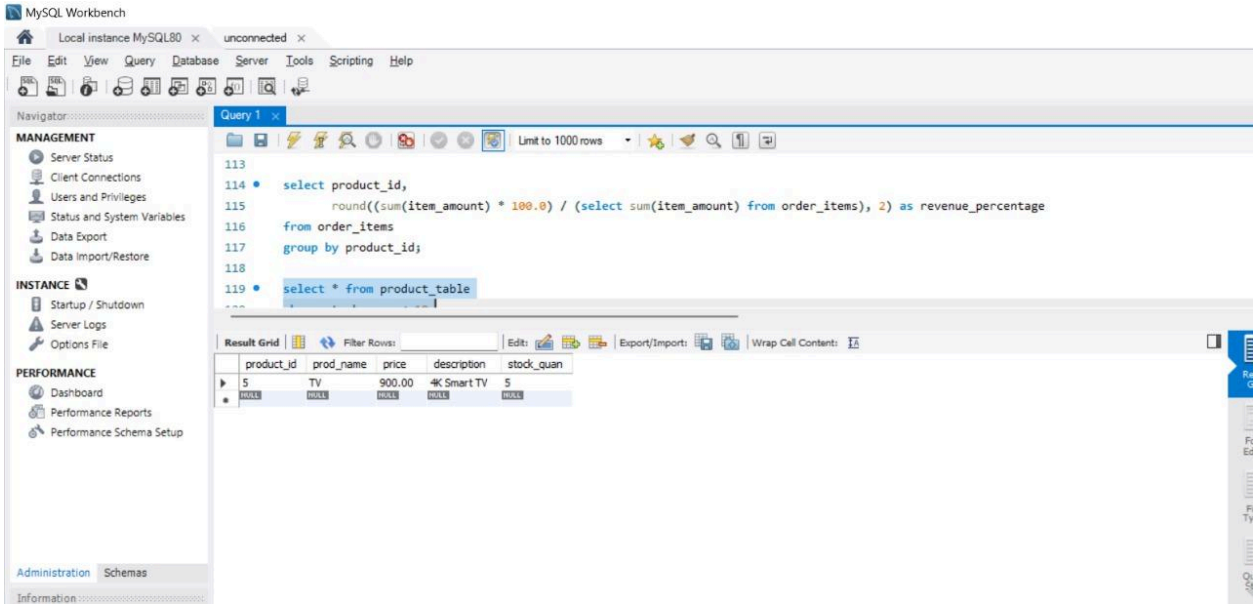
9) select prod_name, min(stock_quan) as min_quantity from product_table group by prod_name;



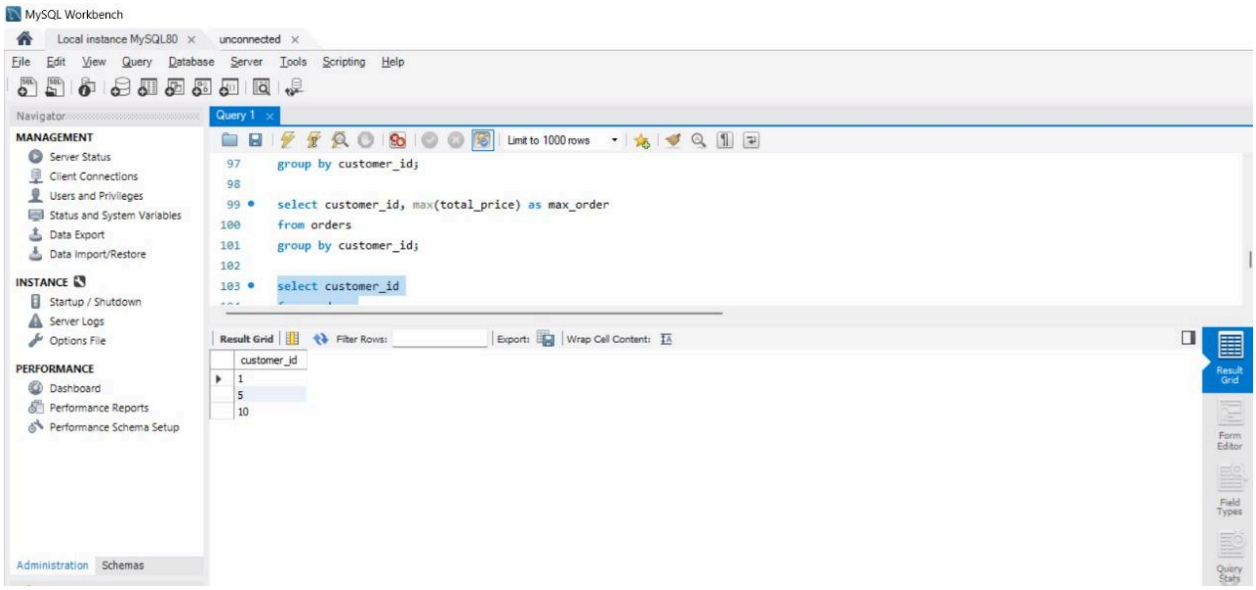
10) select c.customer_id, c.first_name, sum(o.total_price) as total_spenfrom customers c join orders o on c.customer_id = o.customer_id group by c.customer_id;



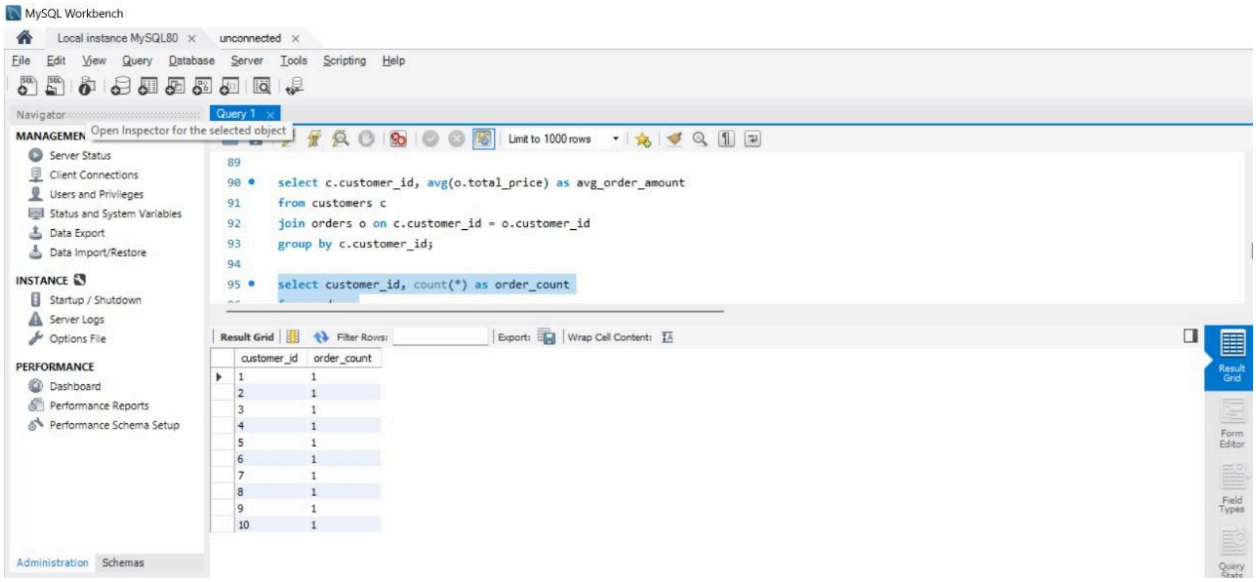
11) select c.customer_id, avg(o.total_price) as avg_order_amount from customers c join orders o on c.customer_id = o.customer_id group by c.customer_id;



12) select customer_id, count(*) as order_count from orders group by customer_id;

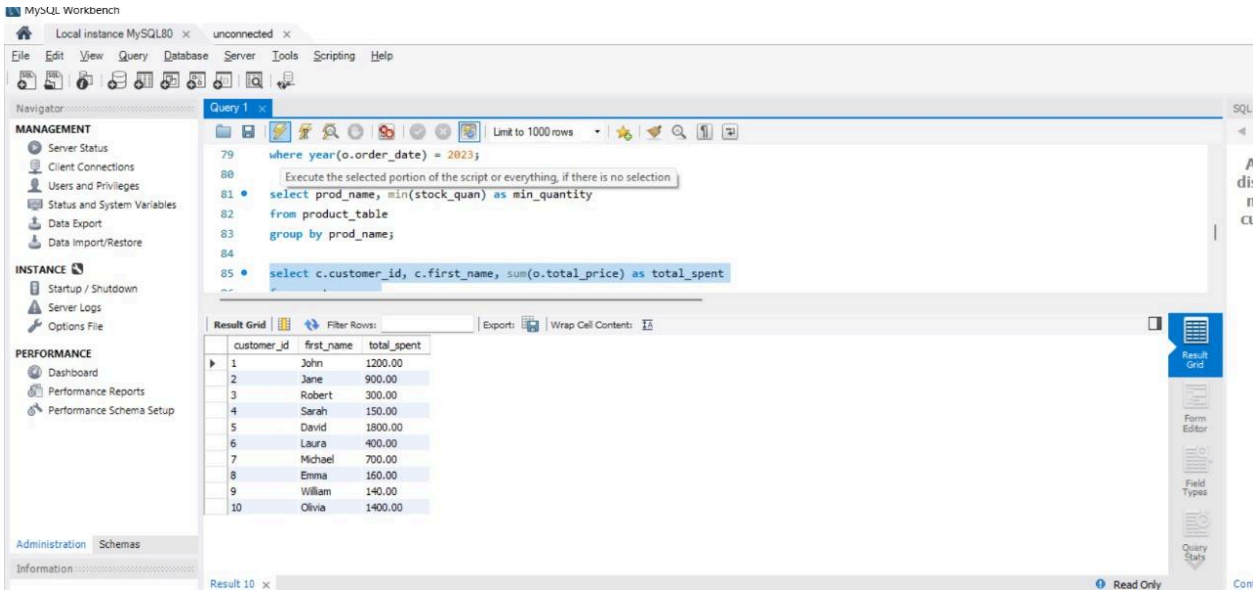


13)select customer_id, max(total_price) as max_order from orders group by customer_id;

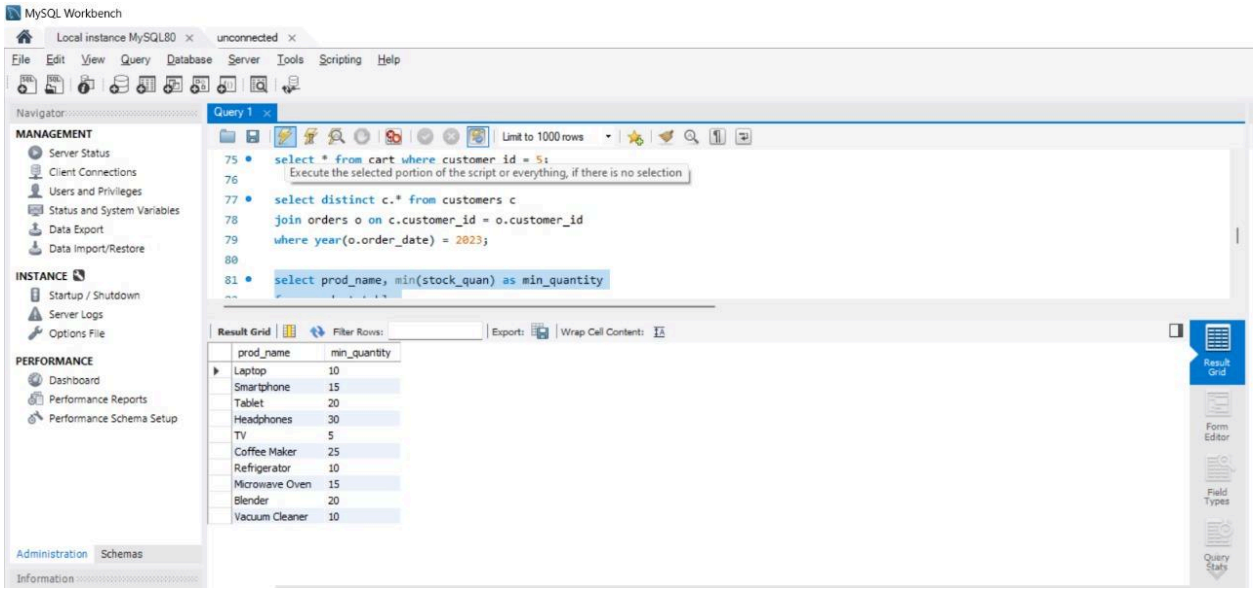


14.

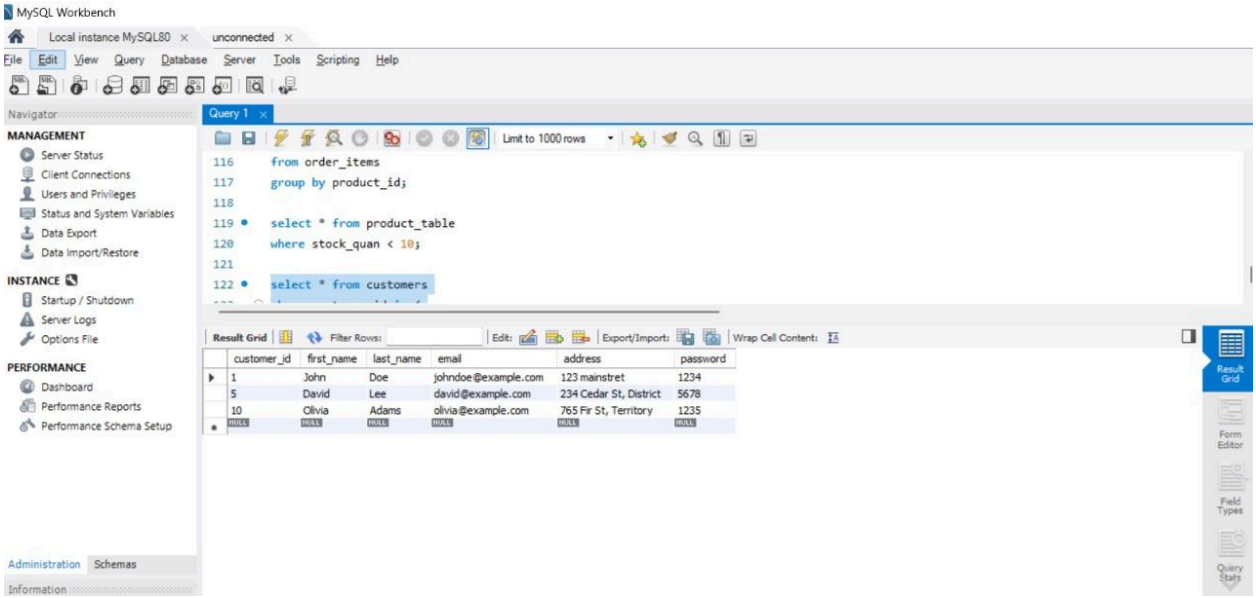
14)select customer_id from orders group by customer_id having sum(total_price) > 1000;



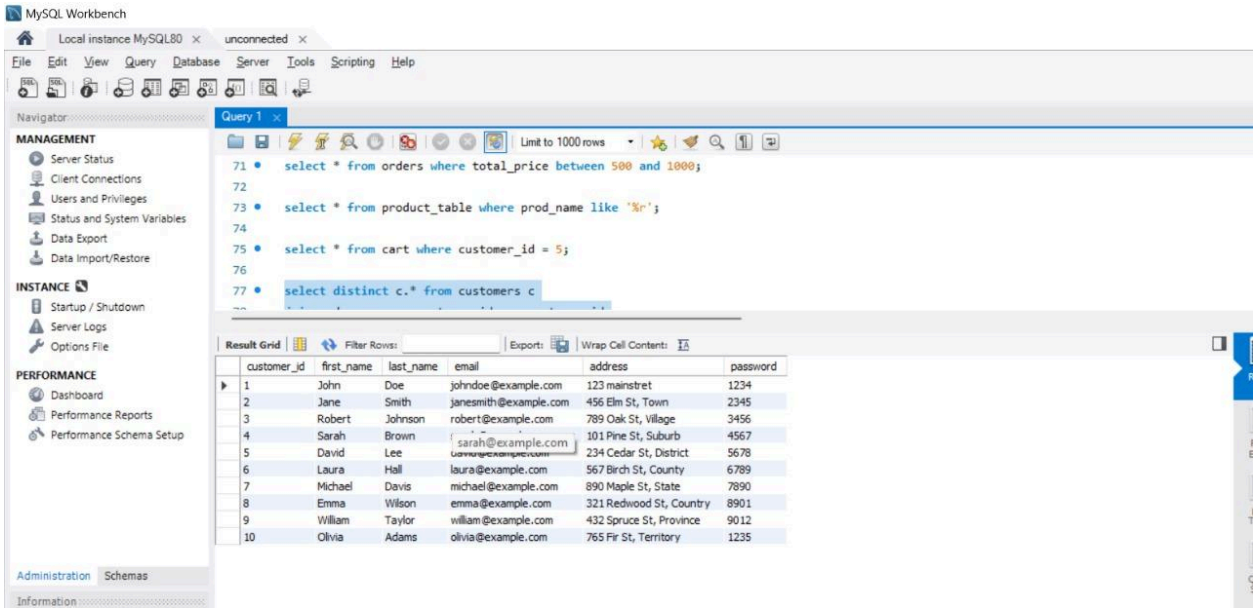
15) select * from product_table where product_id not in (select product_id from cart);



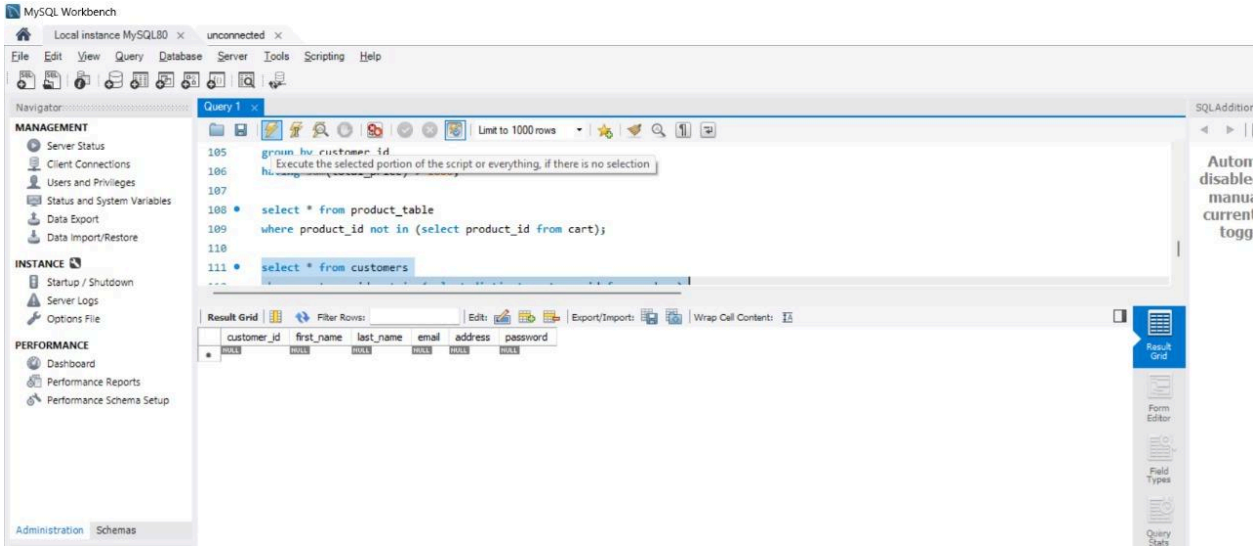
16) select * from customers where customer_id not in (select distinct customer_id from orders);



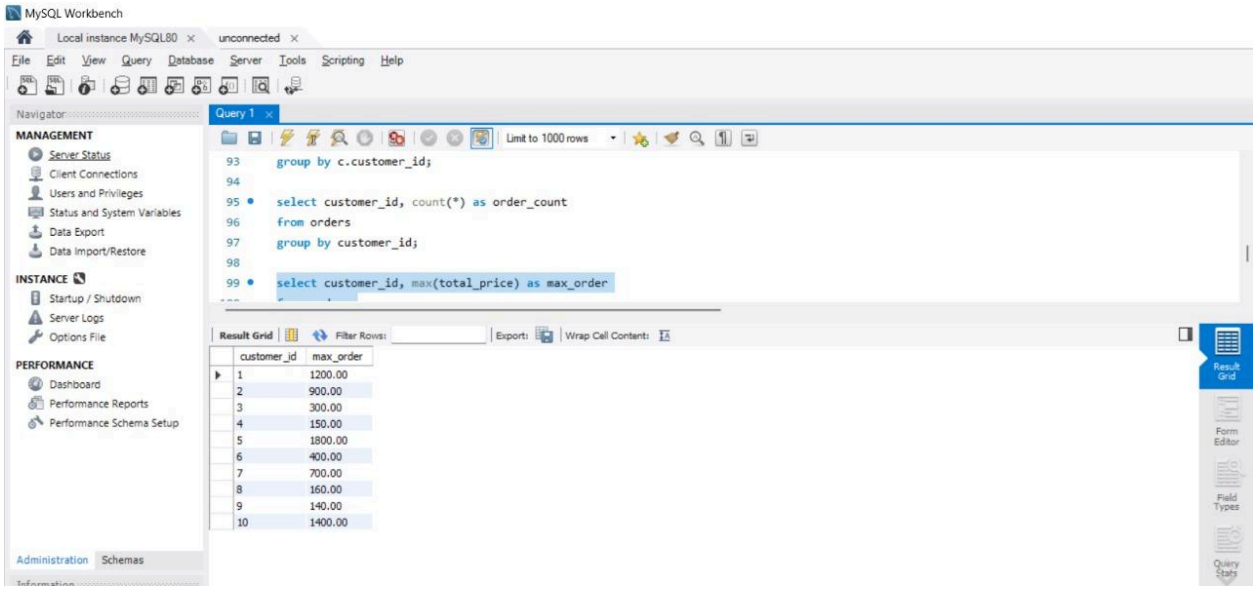
17) select product_id,round((sum(item_amount) * 100.0) / (select sum(item_amount) from order_items), 2) as revenue_percentage from order_items group by product_id;



18) select * from product_table where stock_quan < 10;



19) select * from customers where customer_id in (select customer_id from orders where total_price > 1000);


MySQL Workbench
Local instance MySQL80
unconnected
File Edit View Query Database Server Tools Scripting Help
Navigator
MANAGEMENT
Server Status
Client Connections
Users and Privileges
Status and System Variables
Data Export
Data Import/Restore
INSTANCE
Startup / Shutdown
Server Logs
Options File
PERFORMANCE
Dashboard
Performance Reports
Performance Schema Setup
Administration Schemas
Query 1
Limit to 1000 rows
group by c.customer_id;
select customer_id, count(*) as order_count
from orders
group by customer_id;
select customer_id, max(total_price) as max_order
Result Grid
Filter Rows:
Export:
Wrap Cell Content:
customer_id max_order
1 1200.00
2 900.00
3 300.00
4 150.00
5 1800.00
6 400.00
7 700.00
8 160.00
9 140.00
10 1400.00
Result Grid
Form Editor
Field Types
Query Stats