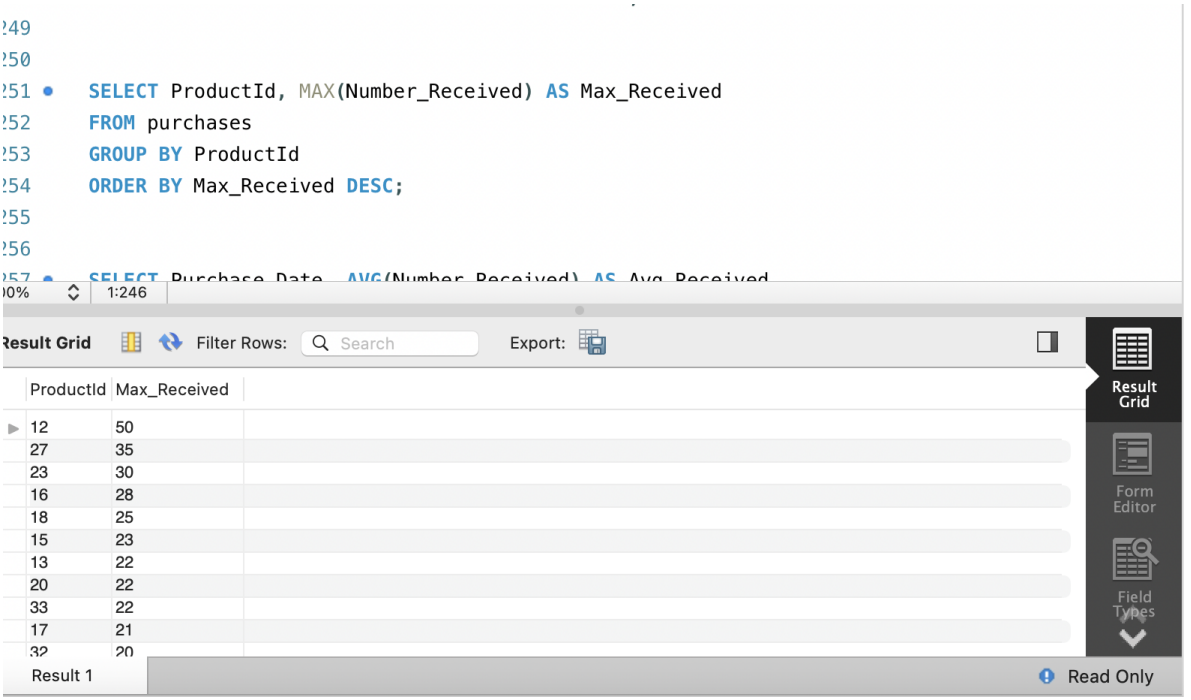
Untitled

1) Find the products with the highest number of units received

SELECT ProductId, MAX(Number_Received) AS Max_Received
FROM purchases
GROUP BY ProductId
ORDER BY Max_Received DESC;



2)Find the total quantity of products ordered in each order, and the associated product and order.

SELECT o.OrderId, o.Order_Date, p.ProductName, SUM(pu.Number_Received) AS TotalOrdered
FROM orders o
JOIN purchases pu ON o.ProductId = pu.ProductId
JOIN products p ON o.ProductId = p.id
GROUP BY o.OrderId, o.Order_Date, p.ProductName
ORDER BY o.OrderId;

```
259     GROUP BY Purchase_Date;
260
261
262
263 •   SELECT o.OrderId, o.Order_Date, p.ProductName, SUM(pu.Number_Received) AS TotalOrdered
264     FROM orders o
265     JOIN purchases pu ON o.ProductId = pu.ProductId
266     JOIN products p ON o.ProductId = p.id
267     GROUP BY o.OrderId, o.Order_Date, p.ProductName
268     ORDER BY o.OrderId;
269
270
271 •   UPDATE products
```

100% 1:260

Result Grid | Filter Rows: Search | Export:

OrderId	Order_Date	...	TotalOrdered

Result 2 | Read Only

3)To know the top spender of products?

```
SELECT p.ProductName, SUM(o.Number_Shipped) AS TotalOrders
FROM products p
LEFT JOIN orders o ON p.id = o.ProductId
GROUP BY p.ProductName
ORDER BY TotalOrders DESC
LIMIT 1;
```

```
347
348
349 •   SELECT p.ProductName, SUM(o.Number_Shipped) AS TotalOrders
350     FROM products p
351     LEFT JOIN orders o ON p.id = o.ProductId
352     GROUP BY p.ProductName
353     ORDER BY TotalOrders DESC
354     LIMIT 1;
355
356
357
358
```

100% 1:357

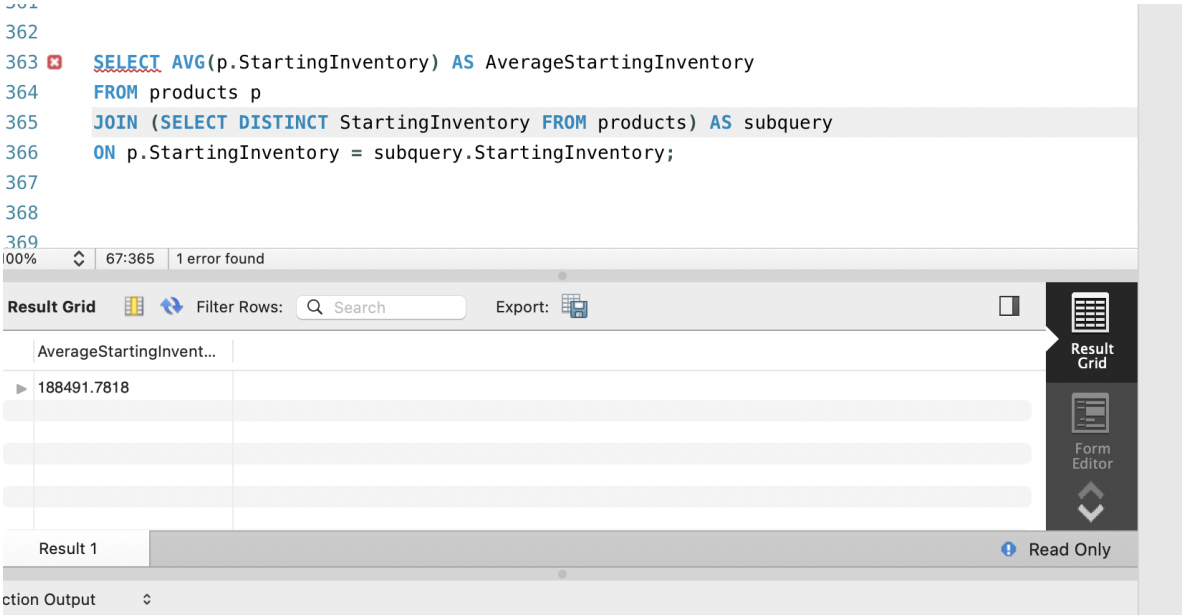
Result Grid | Filter Rows: Search | Export: | Fetch rows:

...	Tot...
HI	NULL

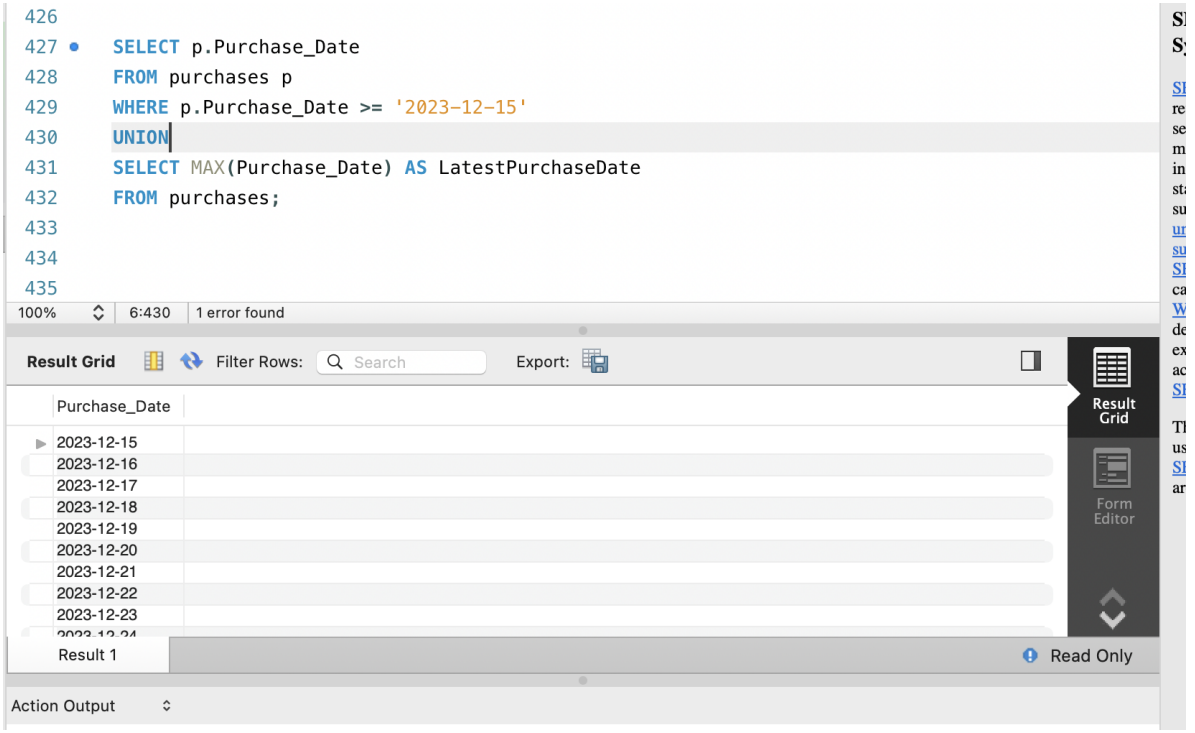
Result 1 | Read Only

4)Calculate the average starting inventory of all products.

SELECT AVG(p.StartingInventory) AS AverageStartingInventory
FROM products p
JOIN (SELECT DISTINCT StartingInventory FROM products) AS subquery
ON p.StartingInventory = subquery.StartingInventory;

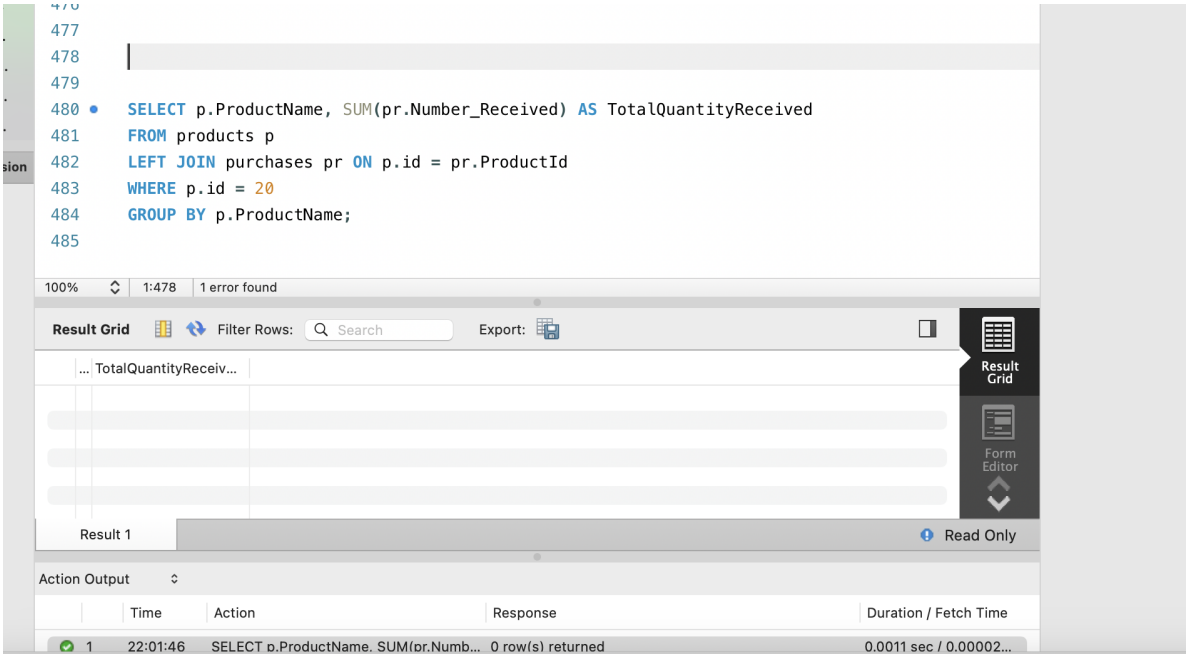


5)List the purchases made on or after 2023-12-15.Find the latest purchase date in the table.
SELECT p.Purchase_Date
FROM purchases p
WHERE p.Purchase_Date >= '2023-12-15'
UNION
SELECT MAX(Purchase_Date) AS LatestPurchaseDate
FROM purchases;



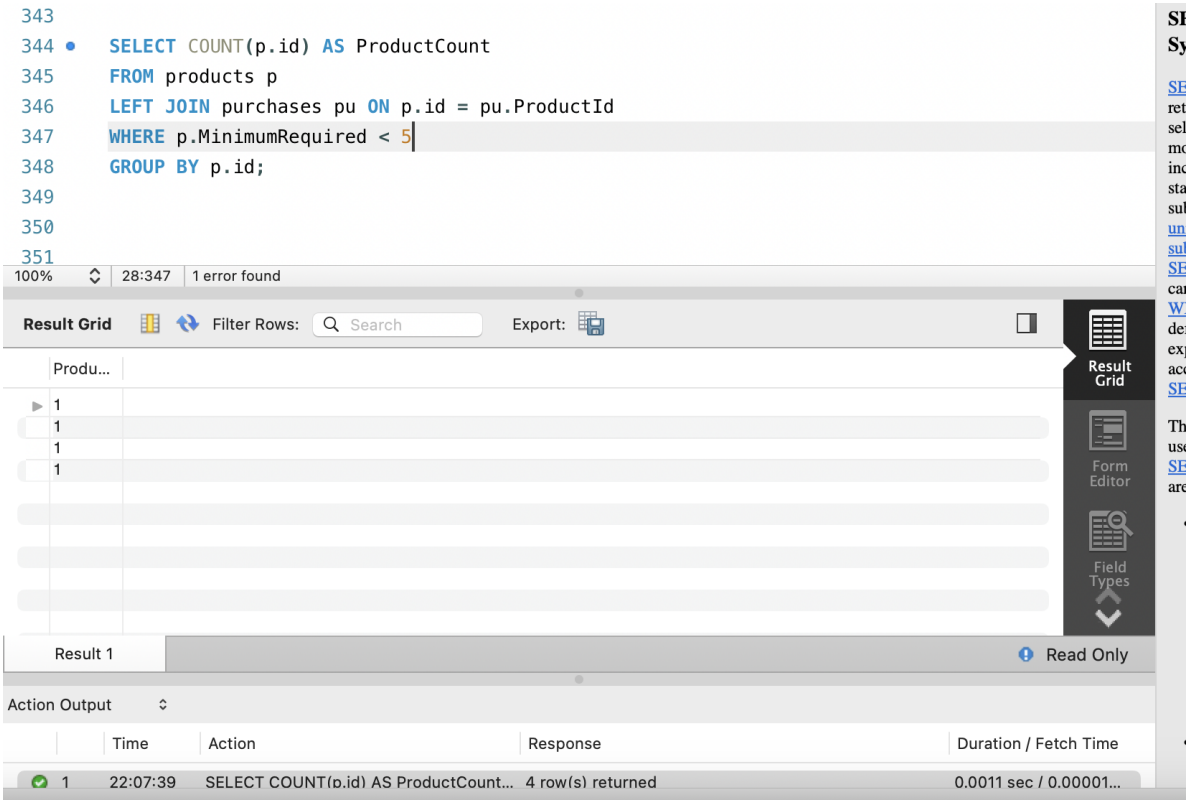
6)Calculate the total quantity received for ProductId 20.

```
SELECT p.ProductName, SUM(pr.Number_Received) AS TotalQuantityReceived
FROM products p
LEFT JOIN purchases pr ON p.id = pr.ProductId
WHERE p.id = 20
GROUP BY p.ProductName;
```



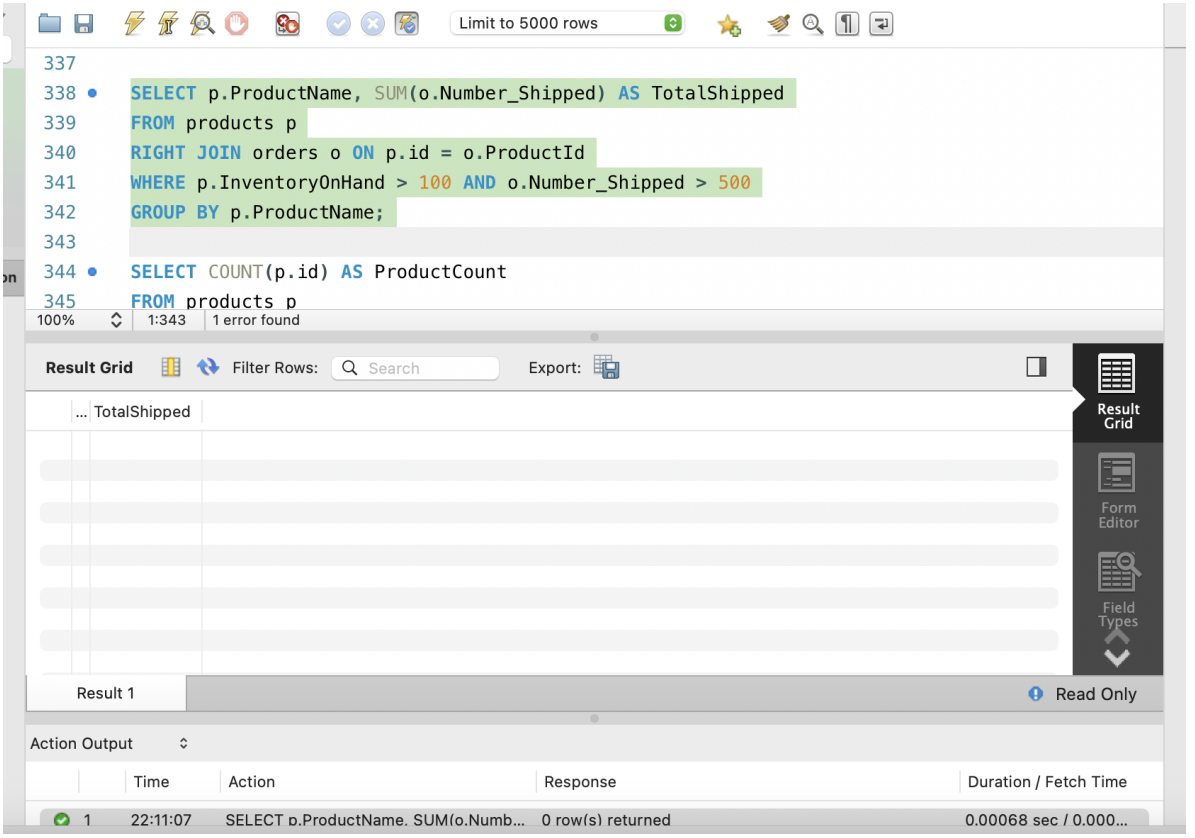
7)How many products have a minimum required quantity of less than 5.

```
SELECT COUNT(p.id) AS ProductCount
FROM products p
LEFT JOIN purchases pu ON p.id = pu.ProductId
WHERE p.MinimumRequired < 5
GROUP BY p.id ;
```



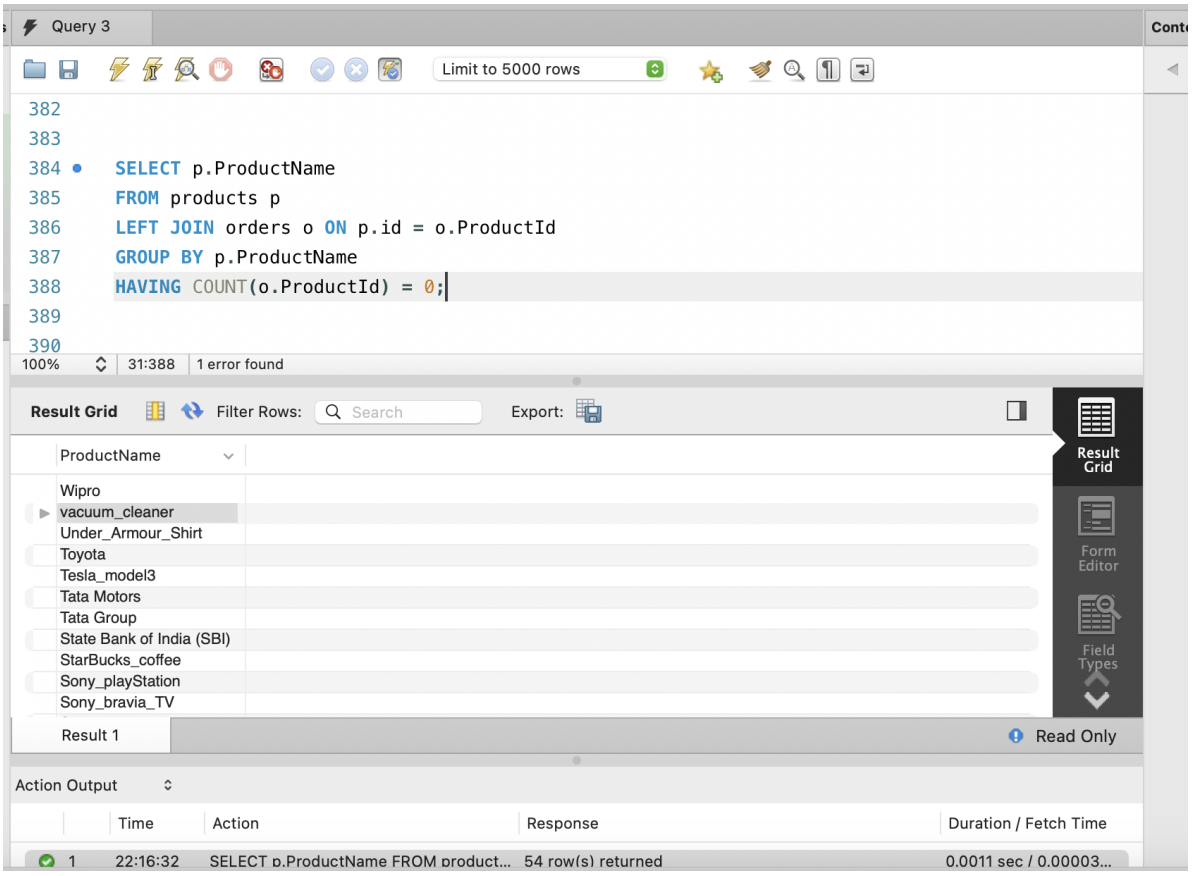
8)Find products with inventory on hand greater than 100 and ordered more than 500.

```
SELECT p.ProductName, SUM(o.Number_Shipped) AS TotalShipped
FROM products p
RIGHT JOIN orders o ON p.id = o.ProductId
WHERE p.InventoryOnHand > 100 AND o.Number_Shipped > 500
GROUP BY p.ProductName;
```



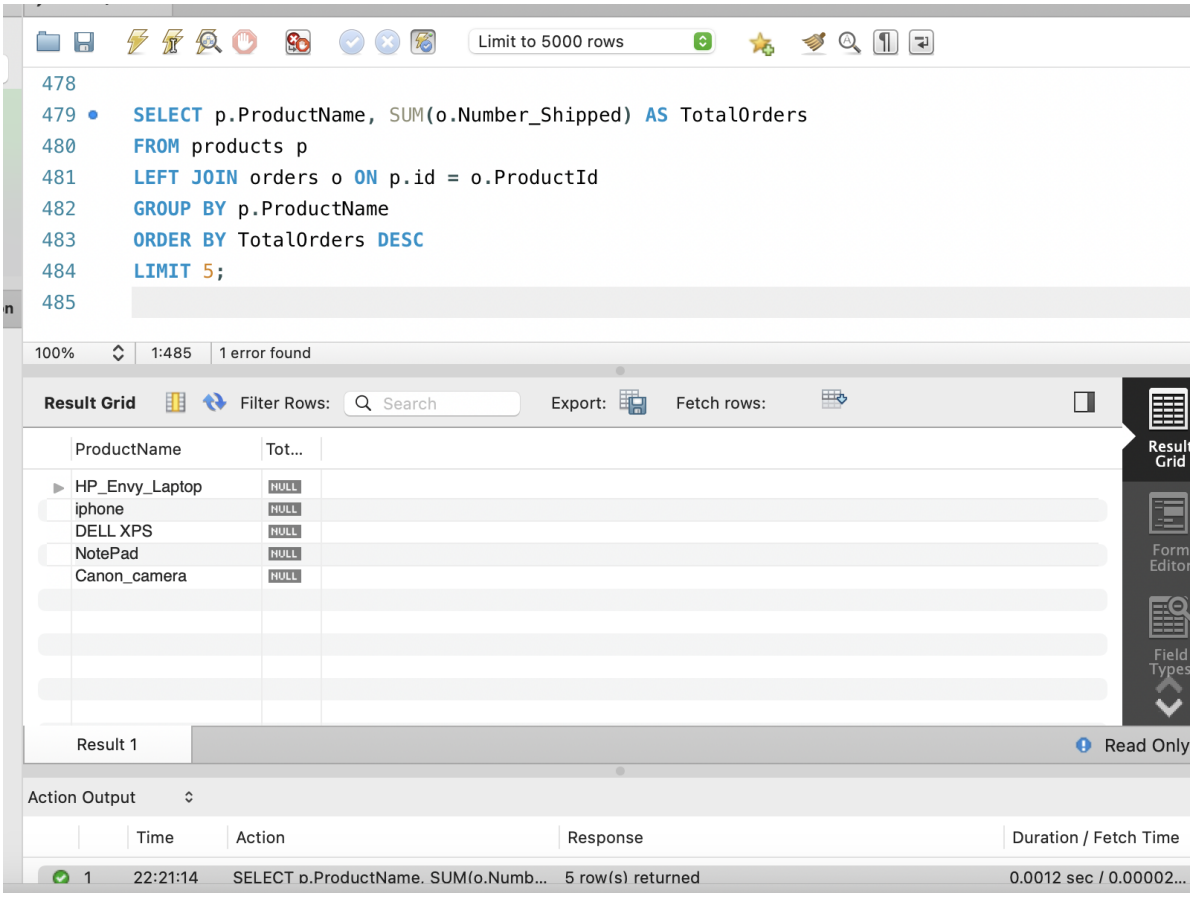
9)Find the products that have not been ordered yet.

```
SELECT p.ProductName
FROM products p
LEFT JOIN orders o ON p.id = o.ProductId
GROUP BY p.ProductName
HAVING COUNT(o.ProductId) = 0;
```



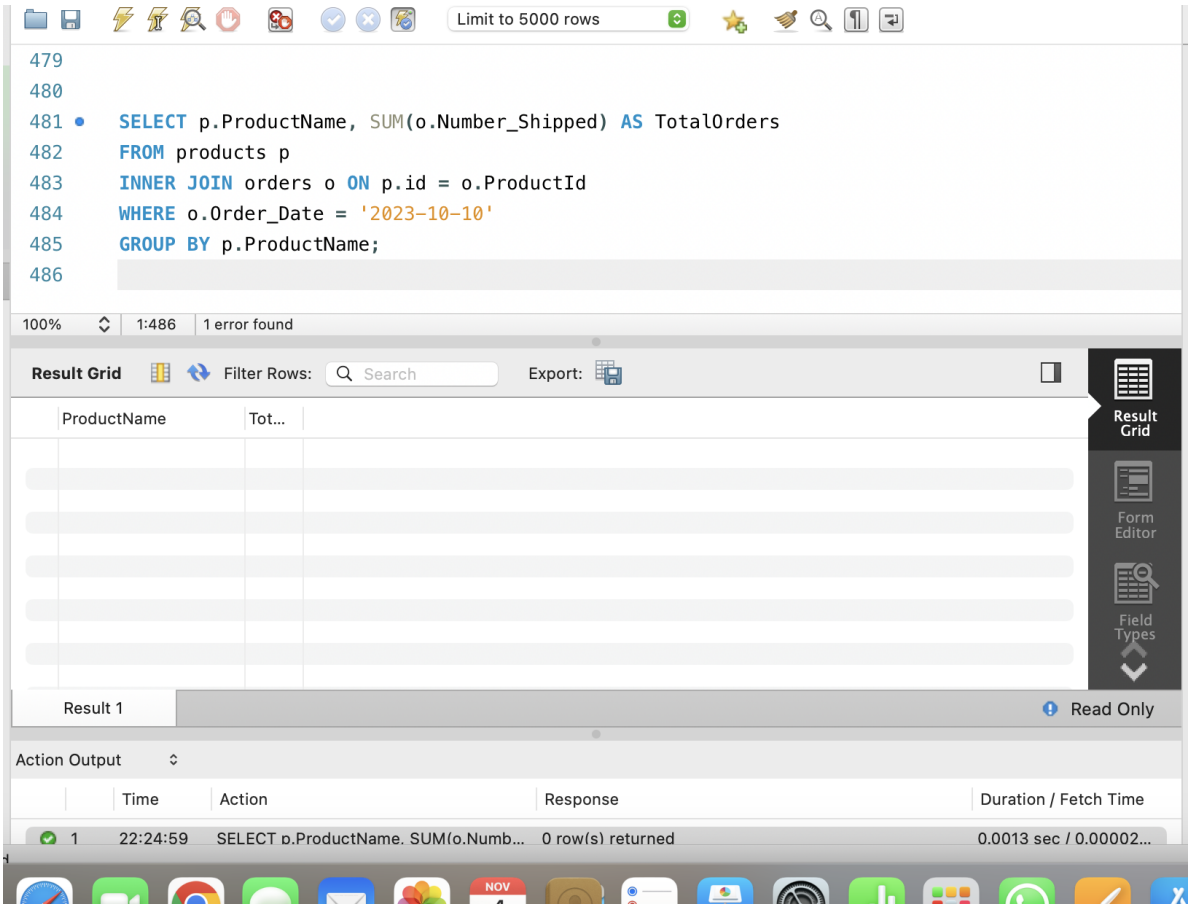
10)The LIMIT 1 ensures that we retrieve only the product with the highest total number of orders.

```
SELECT p.ProductName, SUM(o.Number_Shipped) AS TotalOrders
FROM products p
LEFT JOIN orders o ON p.id = o.ProductId
GROUP BY p.ProductName
ORDER BY TotalOrders DESC
LIMIT 5;
```



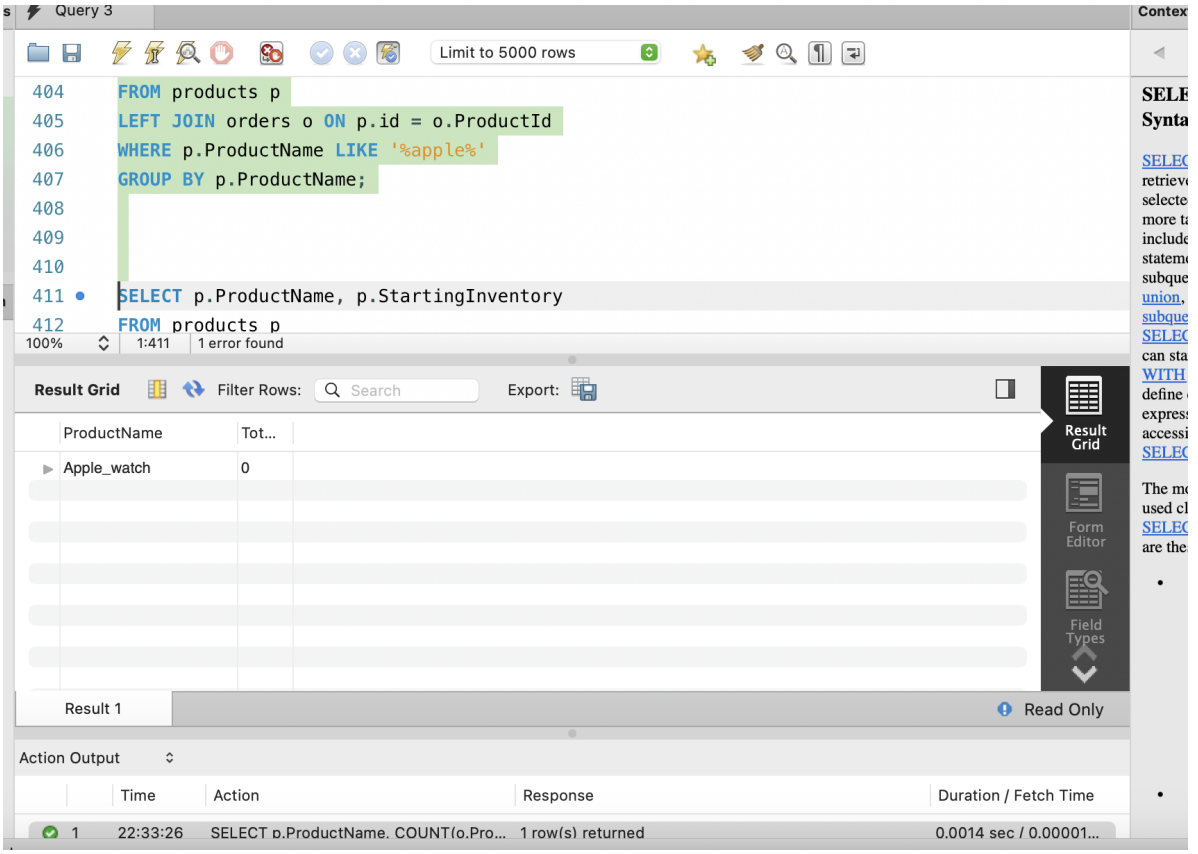
11)Retrieve the products that were ordered on 2023-10-10.

```
SELECT p.ProductName, SUM(o.Number_Shipped) AS TotalOrders
FROM products p
INNER JOIN orders o ON p.id = o.ProductId
WHERE o.Order_Date = '2023-10-10'
GROUP BY p.ProductName;
```



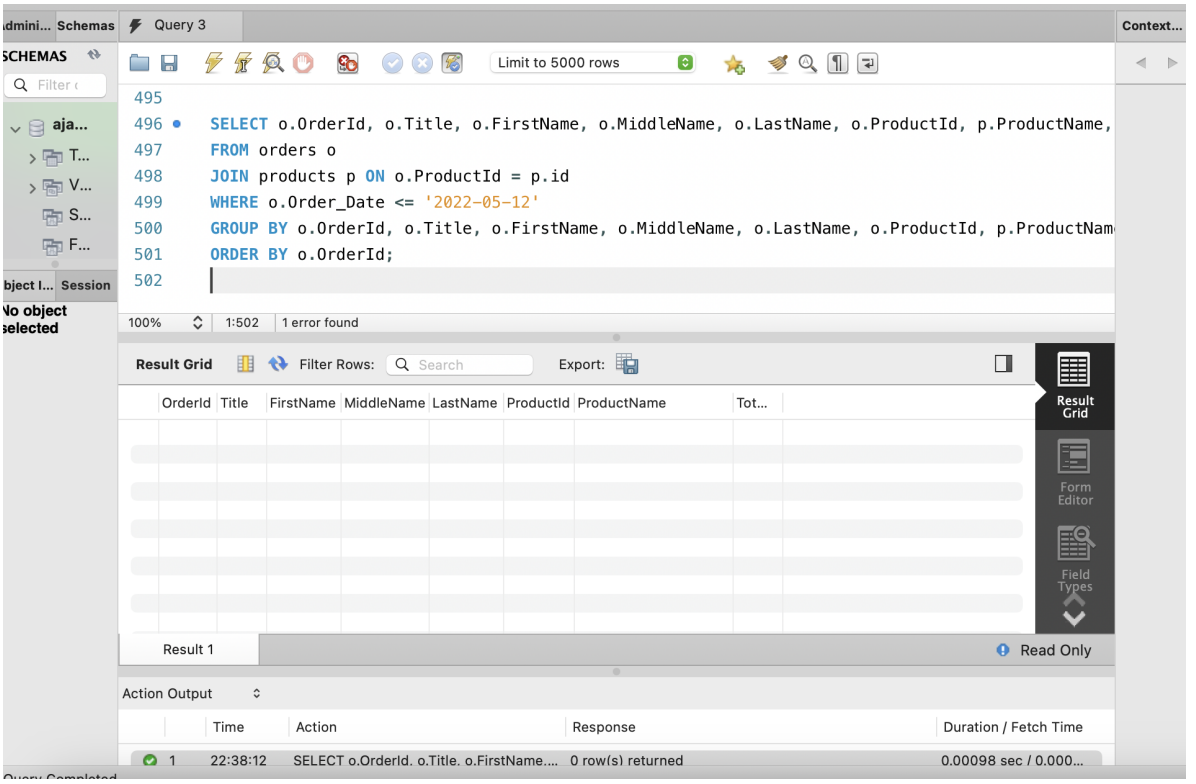
12)List all products with names that contain "apple".

```
SELECT p.ProductName, COUNT(o.ProductId) AS TotalOrders
FROM products p
LEFT JOIN orders o ON p.id = o.ProductId
WHERE p.ProductName LIKE '%apple%'
GROUP BY p.ProductName;
```



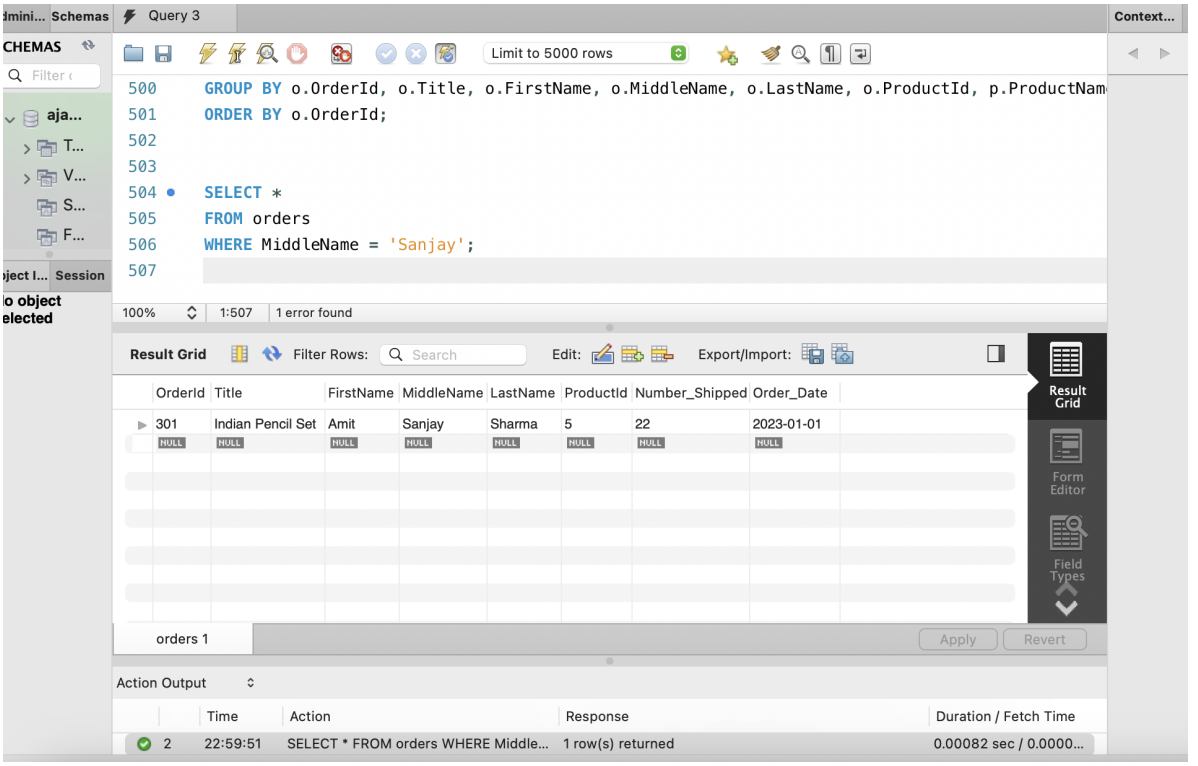
13)Retrieve the orders placed on or before '2022-05-12.'

SELECT o.OrderId, o.Title, o.FirstName, o.MiddleName, o.LastName, o.ProductId, p.ProductName, COUNT(*) AS TotalOrders
FROM orders o
JOIN products p ON o.ProductId = p.id
WHERE o.Order_Date <= '2022-05-12'
GROUP BY o.OrderId, o.Title, o.FirstName, o.MiddleName, o.LastName, o.ProductId, p.ProductName
ORDER BY o.OrderId;



14)Retrieve the orders placed by customers with MiddleName 'Sanjay.'

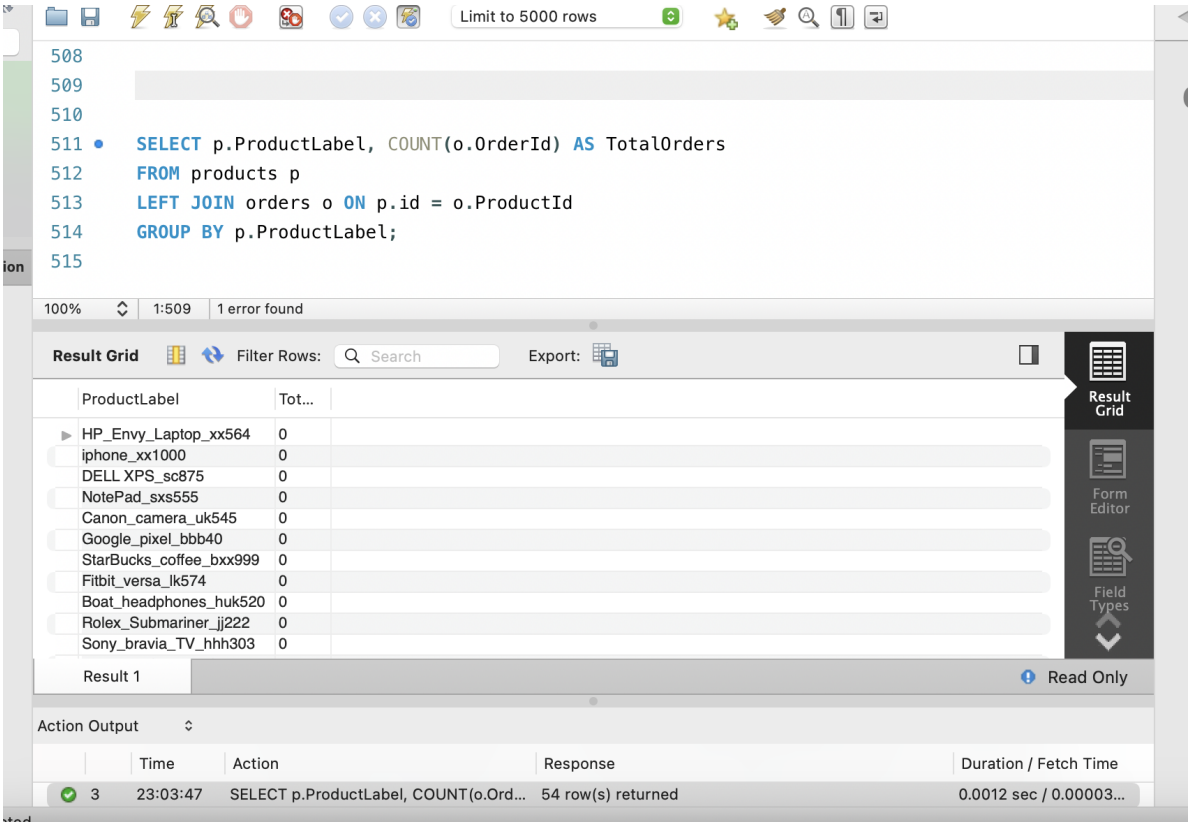
```
SELECT *
FROM orders
WHERE MiddleName = 'Sanjay';
```



15)

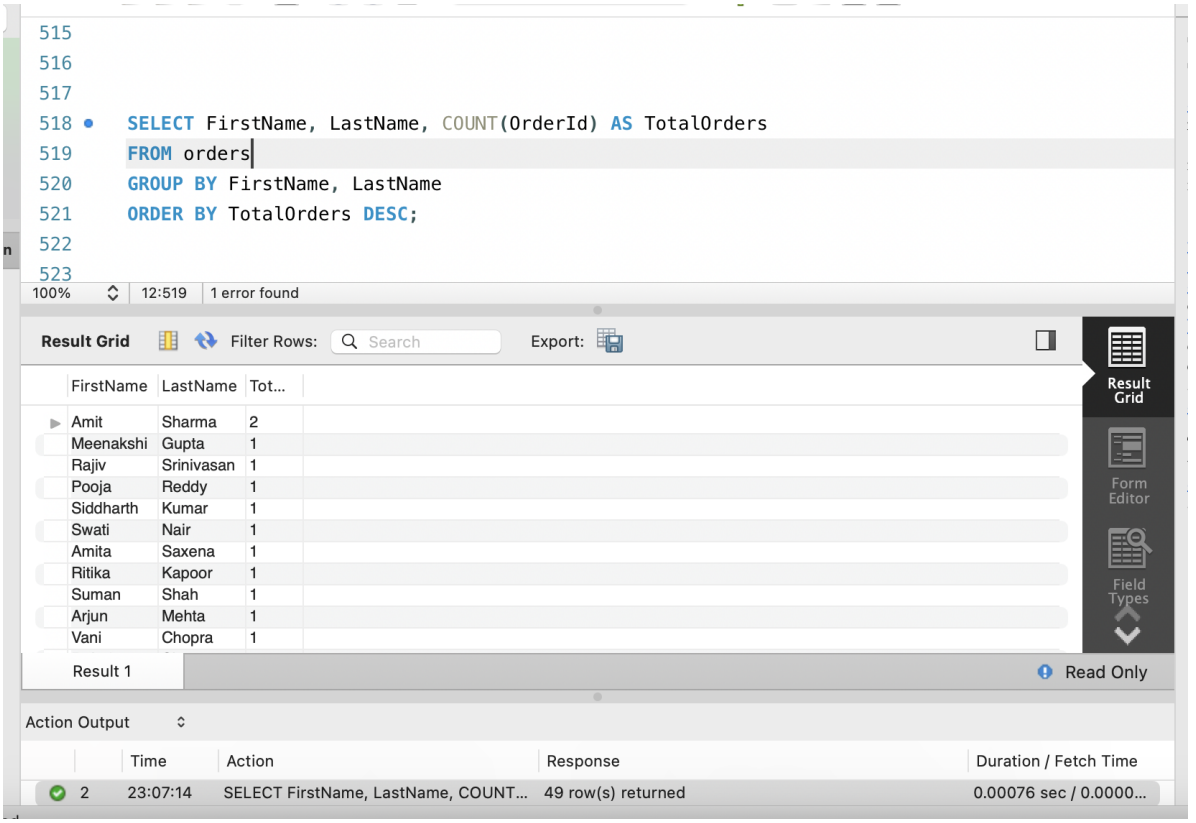
Calculate the total number of orders for each product label (grouped by ProductLabel).

```
SELECT p.ProductLabel, COUNT(o.OrderId) AS TotalOrders
FROM products p
LEFT JOIN orders o ON p.id = o.ProductId
GROUP BY p.ProductLabel;
```



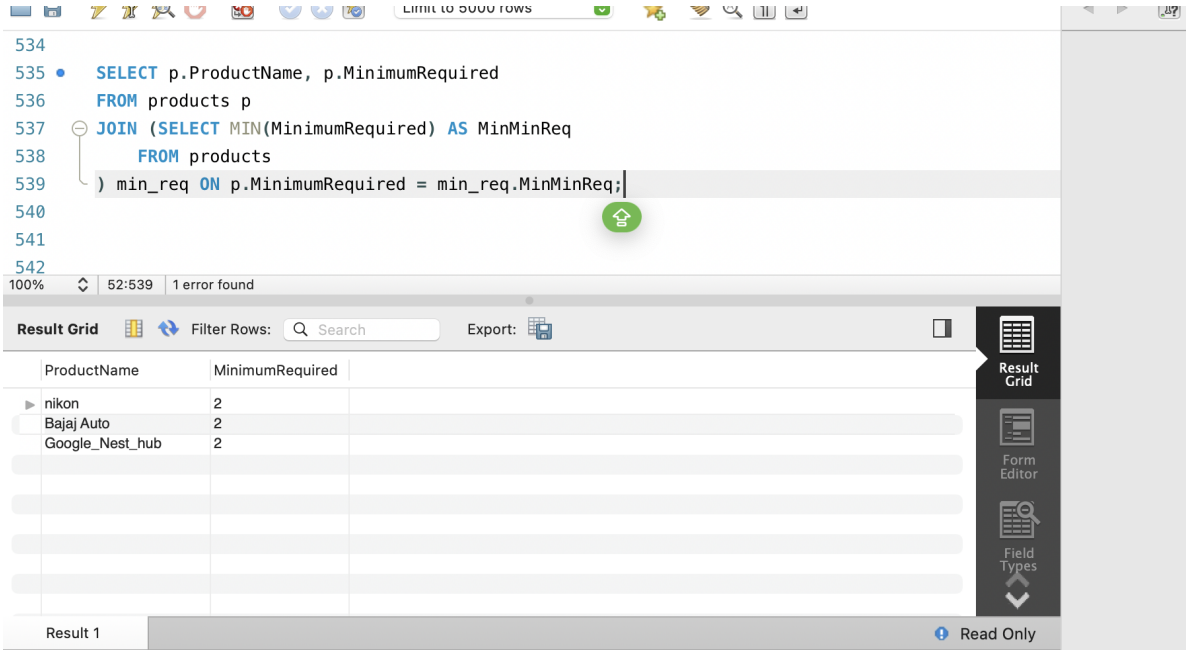
16)Calculate the total number of orders placed by each customer (grouped by FirstName and LastName).

```
SELECT FirstName, LastName, COUNT(OrderId) AS TotalOrders
FROM orders
GROUP BY FirstName, LastName
ORDER BY TotalOrders DESC;
```



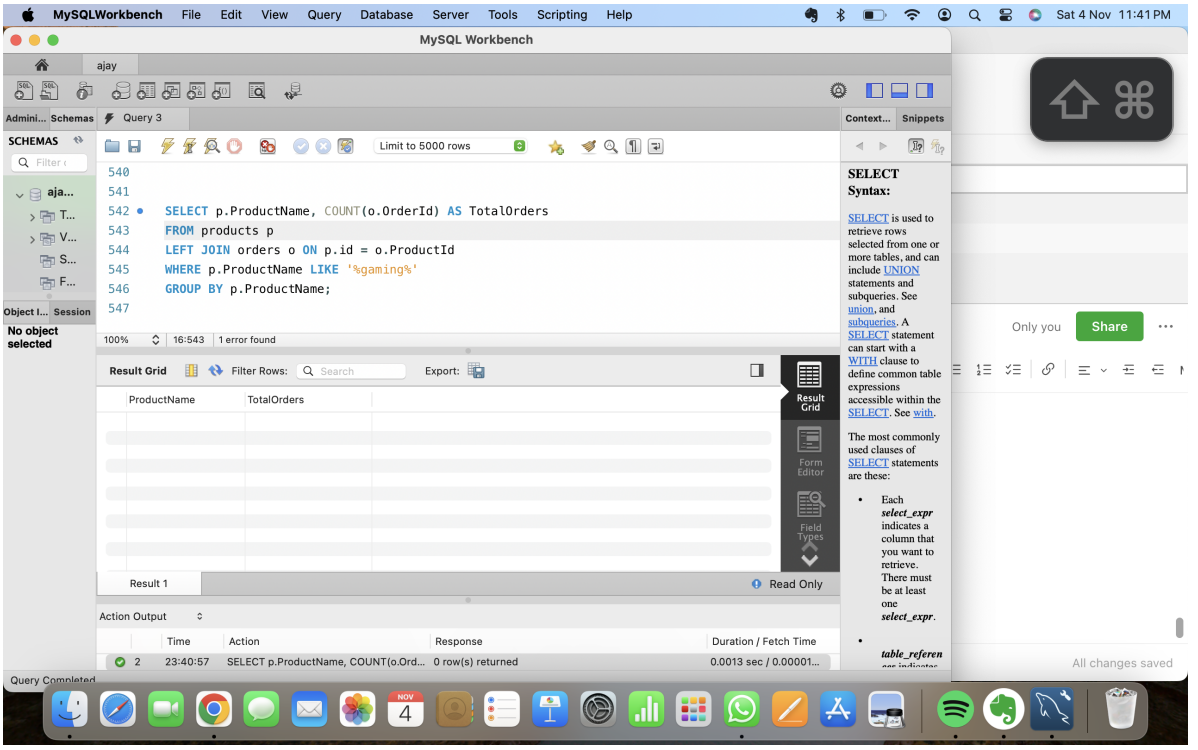
17)Find the products with the lowest minimum required quantity.

```
SELECT p.ProductName, p.MinimumRequired
FROM products p
JOIN (SELECT MIN(MinimumRequired) AS MinMinReq
    FROM products
) min_req ON p.MinimumRequired = min_req.MinMinReq;
```



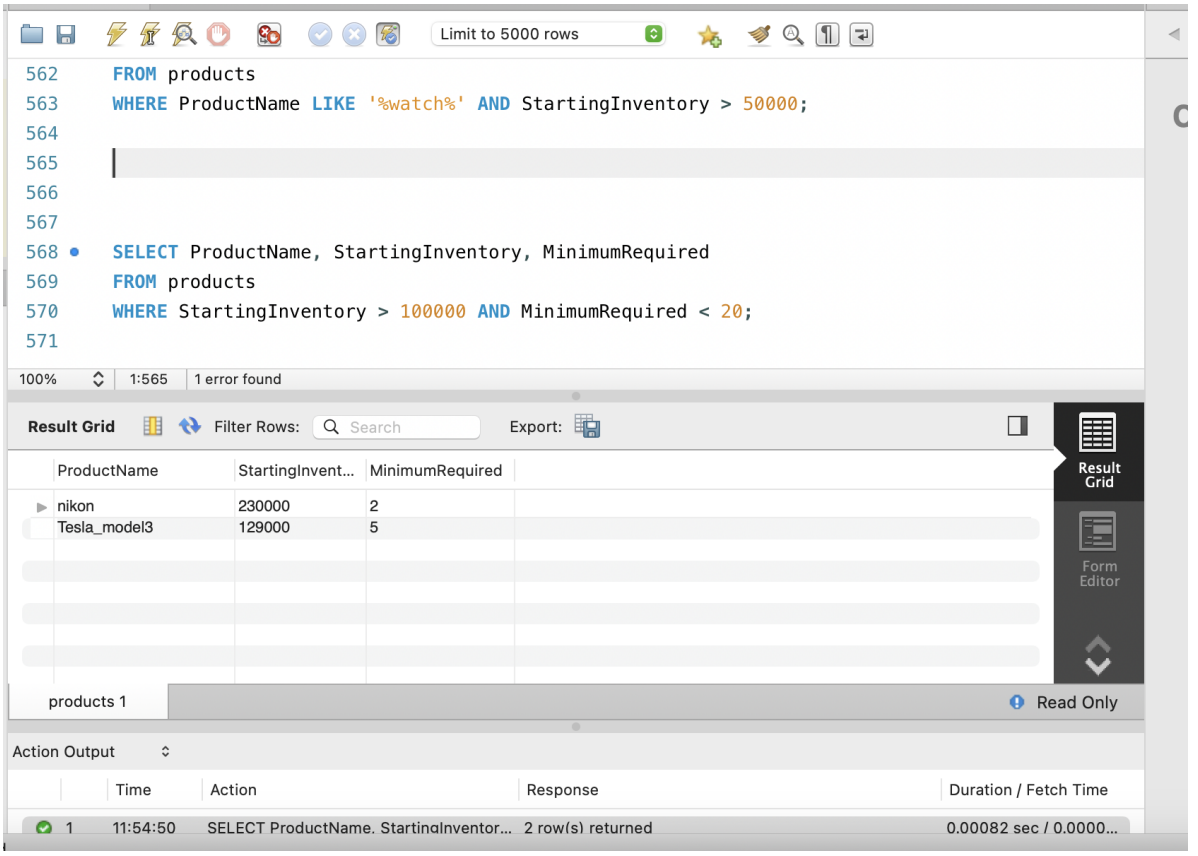
18)List all products with names containing "gaming".

```
SELECT p.ProductName, COUNT(o.OrderId) AS TotalOrders
FROM products p
LEFT JOIN orders o ON p.id = o.ProductId
WHERE p.ProductName LIKE '%gaming%'
GROUP BY p.ProductName;
```



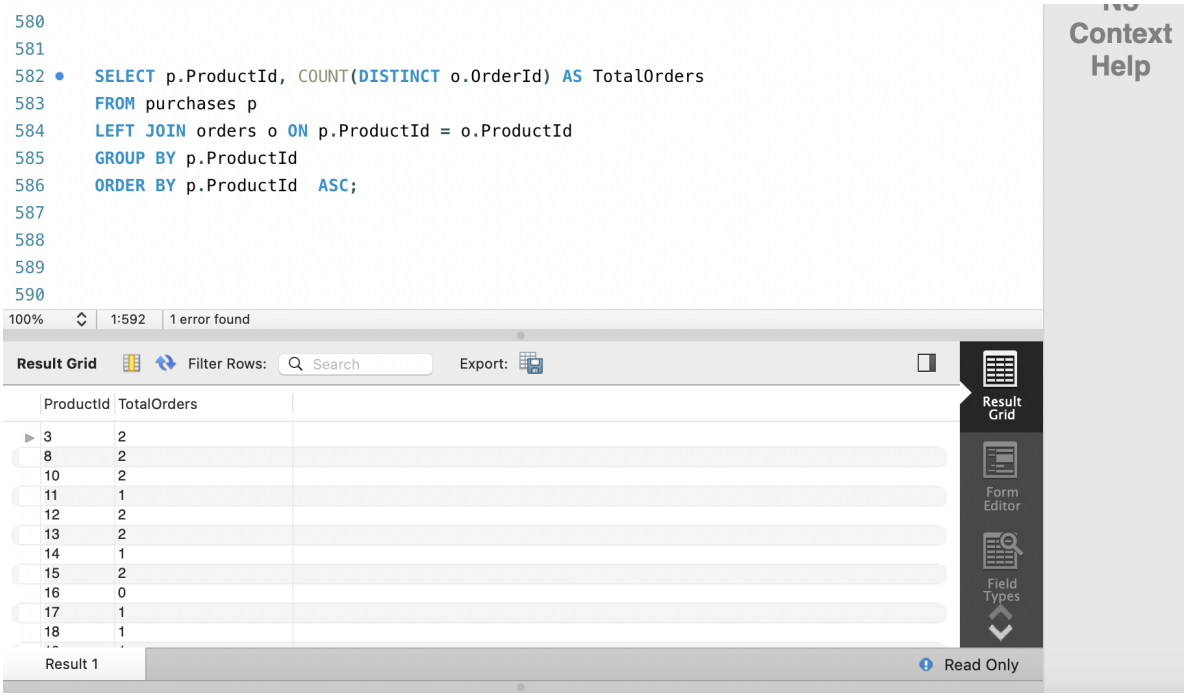
19) Retrieve a list of products that have a 'StartingInventory' greater than 100,000 and a 'MinimumRequired' quantity less than 20."

```
SELECT ProductName, StartingInventory, MinimumRequired
FROM products
WHERE StartingInventory > 100000 AND MinimumRequired < 20;
```



20) To find out how many orders were made for purchases.

```
SELECT p.ProductId, COUNT(DISTINCT o.OrderId) AS TotalOrders
FROM purchases p
LEFT JOIN orders o ON p.ProductId = o.ProductId
GROUP BY p.ProductId
ORDER BY p.ProductId  ASC;
```



21)Retrieve the 'Title' and 'FirstName' of customers who have placed orders in the year 2023.

SELECT Title, FirstName
FROM orders
WHERE YEAR(Order_Date) BETWEEN 2010 AND 2023
GROUP BY Title, FirstName;

```
592     GROUP BY Title, FirstName;
593
594
595     SELECT Title, FirstName
596     FROM orders
597     WHERE YEAR(Order_Date) BETWEEN 2010 AND 2023
598     GROUP BY Title, FirstName;
599
600
```

100% | 12:596 | 1 error found

Result Grid | Filter Rows: Search | Export:

Title	FirstName
Indian Pencil Set	Amit
Indian Phone Accessories	Meenakshi
Spices and Herbs India	Rajiv
Indian Home Office Essentials	Pooja
Indian Camping Gear	Siddharth
Indian Cooking Spices	Suman
Indian Music Concert	Sangeeta
Professional Art Supplies India	Vedika
Indian Party Decorations	Prisha
Woodworking Tools India	Vivaan
Indian Picnic Gear	Shivansh

Result Grid | Form Editor | Field Types

orders 1 | Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
1	12:46:22	SELECT Title, FirstName FROM orders...	11 row(s) returned	0.00073 sec / 0.0000...