**CREATE DATABASE superstore;**

**USE superstore;**

**Task 1:- Understanding the Data**

**1.** About Data :

The data is about all records of the Superstore. The Table wise information of data are given below:

1. cust\_dimen: Details of all the customers

Customer\_Name (TEXT): Name of the customer

Province (TEXT): Province of the customer

Region (TEXT): Region of the customer

Customer\_Segment (TEXT): Segment of the customer

Cust\_id (TEXT): Unique Customer ID

2. market\_fact: Details of every order item sold

Ord\_id (TEXT): Order ID

Prod\_id (TEXT): Prod ID

Ship\_id (TEXT): Shipment ID

Cust\_id (TEXT): Customer ID

Sales (DOUBLE): Sales from the Item sold

Discount (DOUBLE): Discount on the Item sold

Order\_Quantity (INT): Order Quantity of the Item sold

Profit (DOUBLE): Profit from the Item sold

Shipping\_Cost (DOUBLE): Shipping Cost of the Item sold

Product\_Base\_Margin (DOUBLE): Product Base Margin on the Item sold

3. orders\_dimen: Details of every order placed

Order\_ID (INT): Order ID

Order\_Date (TEXT): Order Date

Order\_Priority (TEXT): Priority of the Order

Ord\_id (TEXT): Unique Order ID

4. prod\_dimen: Details of product category and sub-category

Product\_Category (TEXT): Product Category

Product\_Sub\_Category (TEXT): Product Sub Category

Prod\_id (TEXT): Unique Product ID

5. shipping\_dimen: Details of shipping of orders

Order\_ID (INT): Order ID

Ship\_Mode (TEXT): Shipping Mode

Ship\_Date (TEXT): Shipping Date

Ship\_id (TEXT): Unique Shipment ID

**2.** Relationships and List of Primary key and Foreign Keys:

Relationship of the data:

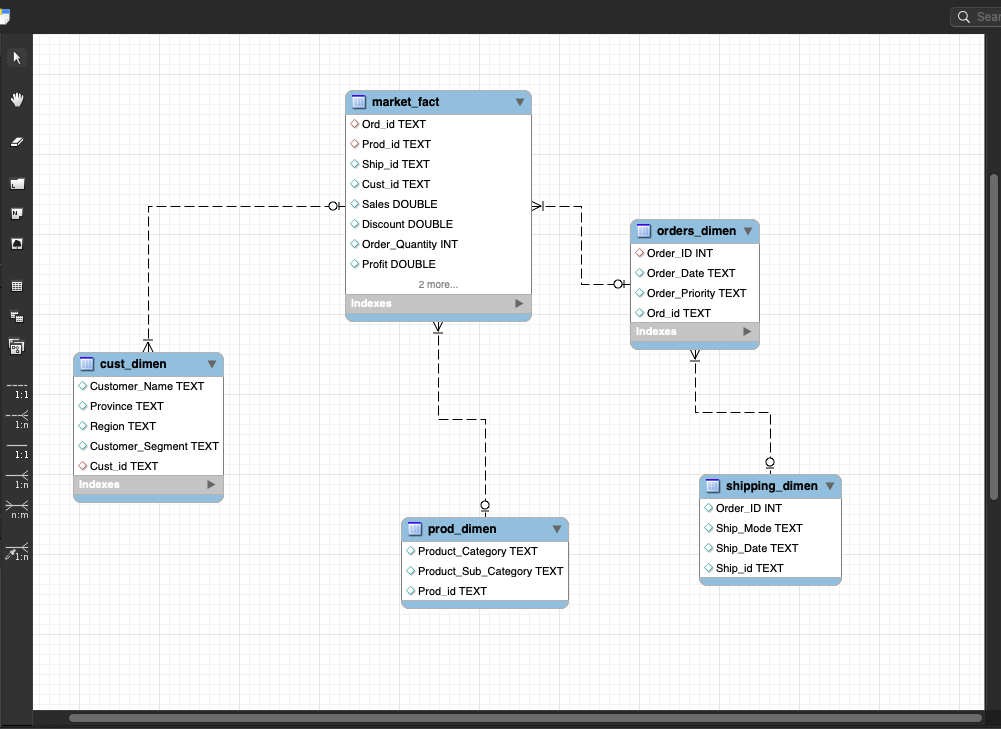
****

Table wise List of Keys:

1. Cust\_dimen:

Primary Key: Cust\_id

Foreign Key: N/A

2. market\_fact:

Primary Key: N/A

Foreign Key: Ord\_id, Prod\_id, Ship\_id, Cust\_id

3. orders\_dimen:

Primary Key: Ord\_id

Foreign Key: N/A

4. prod\_dimen:

Primary Key: Prod\_id, Product\_Sub\_Category

Foreign Key: N/A

5. shipping\_dimen:

Primary Key: Ship\_id

Foreign Key: N/A

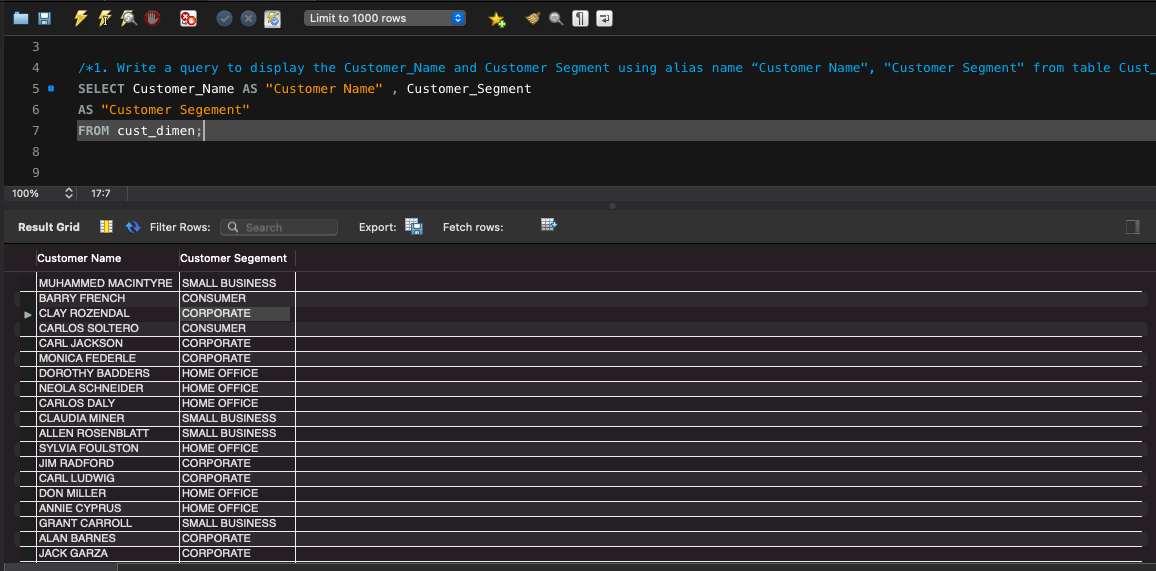
**Task 2:- Basic & Advanced Analysis**

**/\*1. Write a query to display the Customer\_Name and Customer Segment using alias name “Customer Name", "Customer Segment" from table Cust\_dimen.\*/**

SELECT Customer\_Name AS "Customer Name" , Customer\_Segment

AS "Customer Segement"

FROM cust\_dimen;

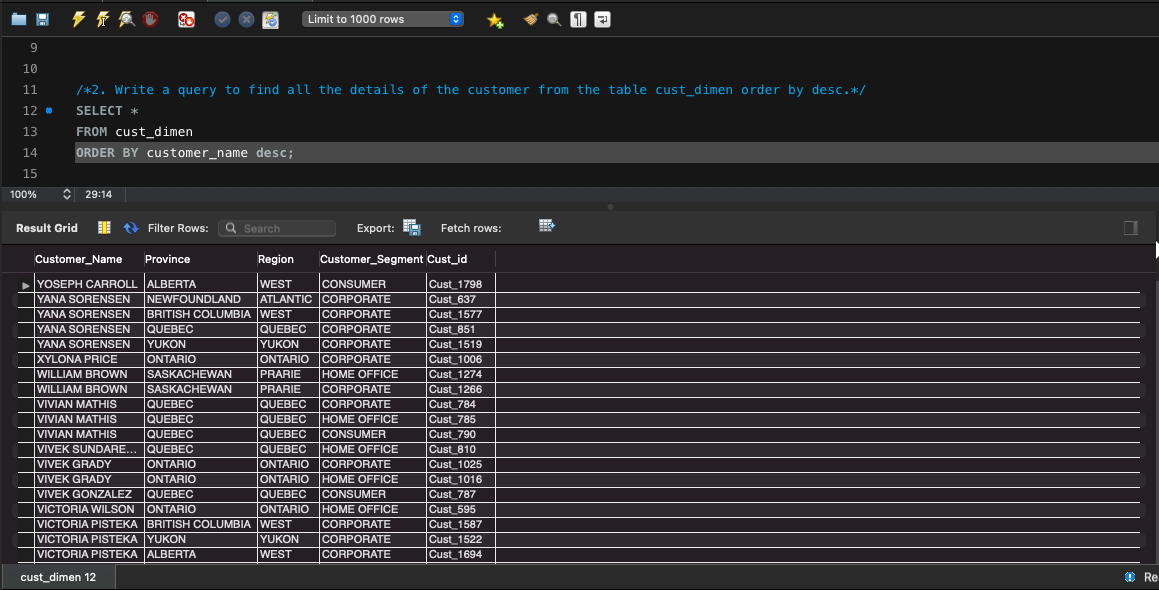


**/\*2. Write a query to find all the details of the customer from the table cust\_dimen order by desc.\*/**

SELECT \*

FROM cust\_dimen

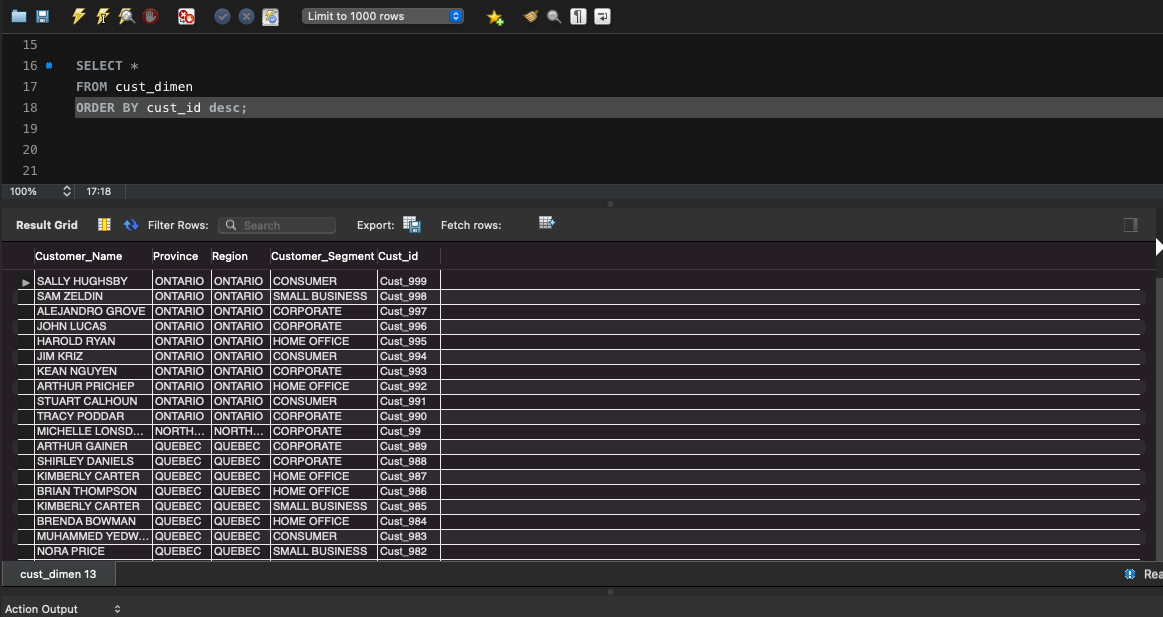
ORDER BY customer\_name desc;



SELECT \*

FROM cust\_dimen

ORDER BY cust\_id desc;

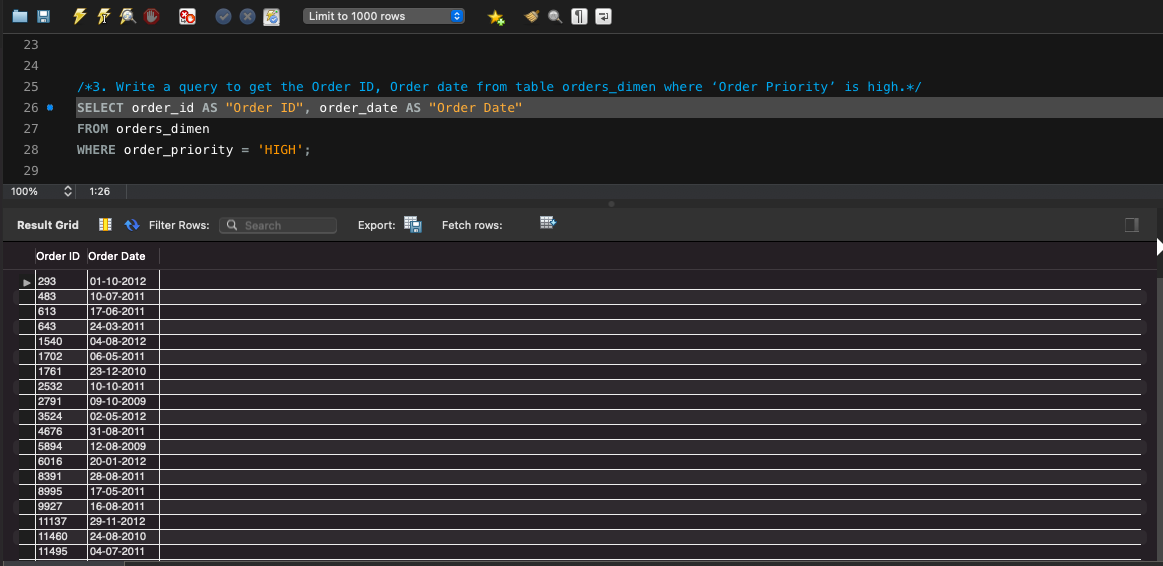


**/\*3. Write a query to get the Order ID, Order date from table orders\_dimen where ‘Order Priority’ is high.\*/**

SELECT order\_id AS "Order ID", order\_date AS "Order Date"

FROM orders\_dimen

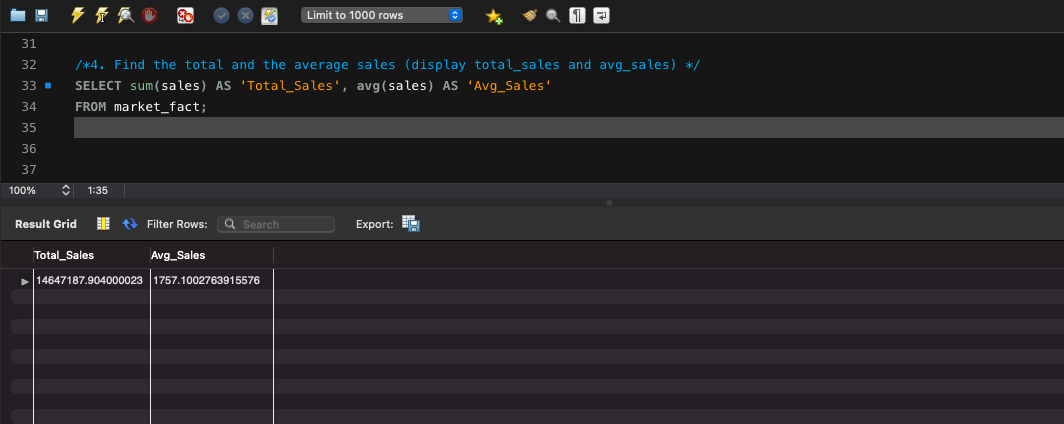
WHERE order\_priority = 'HIGH';



**/\*4. Find the total and the average sales (display total\_sales and avg\_sales) \*/**

SELECT sum(sales) AS 'Total\_Sales', avg(sales) AS 'Avg\_Sales'

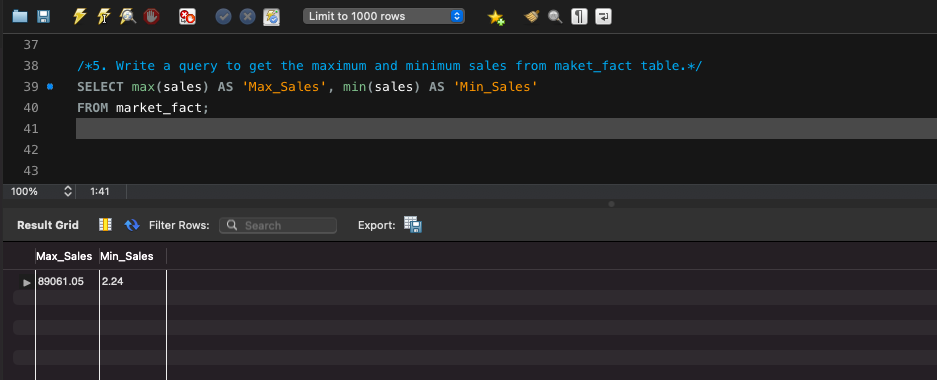
FROM market\_fact;



**/\*5. Write a query to get the maximum and minimum sales from maket\_fact table.\*/**

SELECT max(sales) AS 'Max\_Sales', min(sales) AS 'Min\_Sales'

FROM market\_fact;



**/\*6. Display the number of customers in each region in decreasing order of**

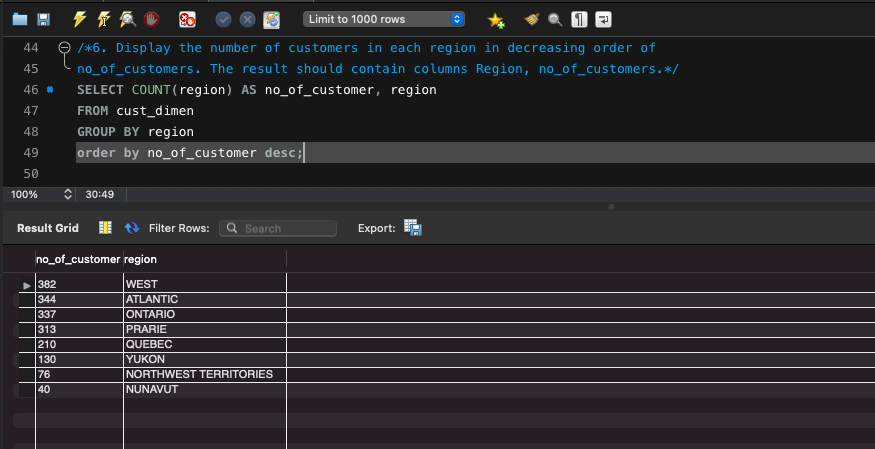
**no\_of\_customers. The result should contain columns Region, no\_of\_customers.\*/**

SELECT COUNT(region) AS no\_of\_customer, region

FROM cust\_dimen

GROUP BY region

order by no\_of\_customer desc;



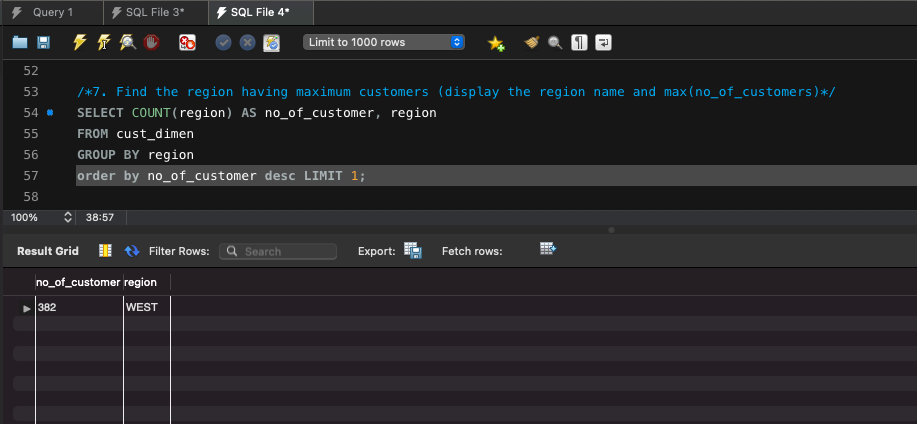
**/\*7. Find the region having maximum customers (display the region name and max(no\_of\_customers)\*/**

SELECT COUNT(region) AS no\_of\_customer, region

FROM cust\_dimen

GROUP BY region

order by no\_of\_customer desc LIMIT 1;



**/\*8. Find all the customers from Atlantic region who have ever purchased ‘TABLES’**

**and the number of tables purchased (display the customer name, no\_of\_tables purchased) \*/**

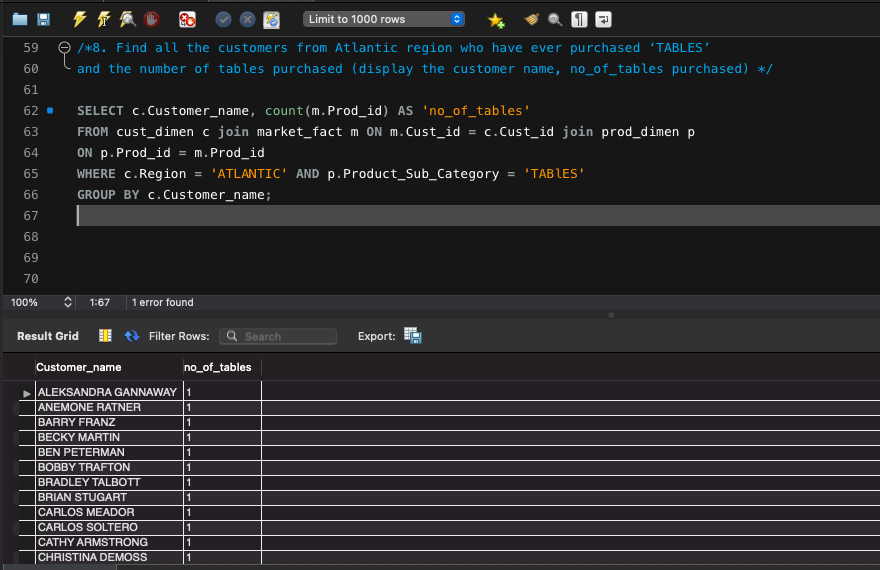
SELECT c.Customer\_name, count(m.Prod\_id) AS 'no\_of\_tables'

FROM cust\_dimen c join market\_fact m ON m.Cust\_id = c.Cust\_id join prod\_dimen p

ON p.Prod\_id = m.Prod\_id

WHERE c.Region = 'ATLANTIC' AND p.Product\_Sub\_Category = 'TABlES'

GROUP BY c.Customer\_name;



**/\*9. Find all the customers from Ontario province who own Small Business. (display**

**the customer name, no of small business owners)\*/**

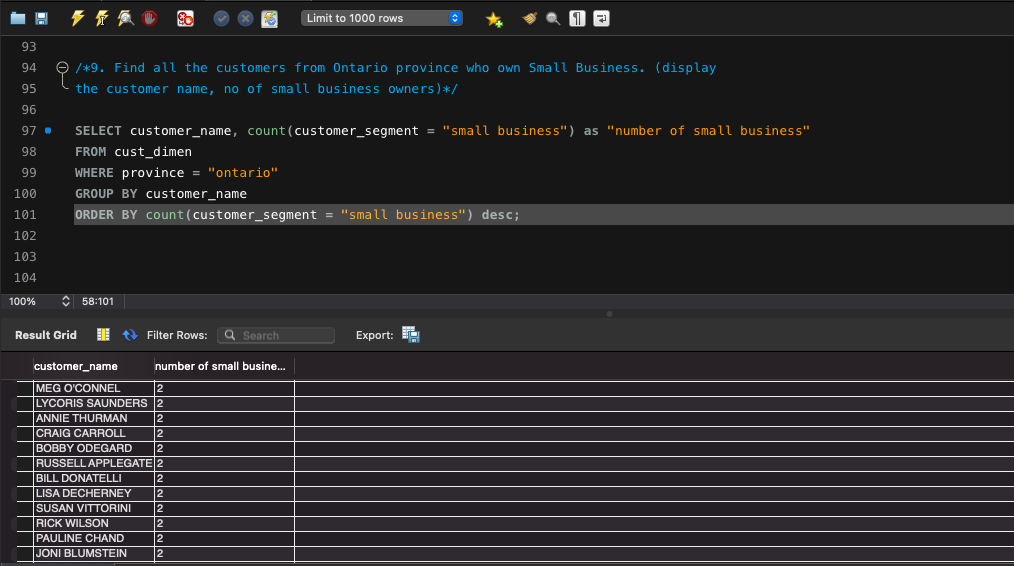
SELECT customer\_name, count(customer\_sagement='SMALL BUSINESS') AS 'number of small business'

FROM cust\_dimen

WHERE province = 'Ontario'

GROUP BY customer\_name

ORDER BY count(customer\_sagement='SMALL BUSINESS') desc;



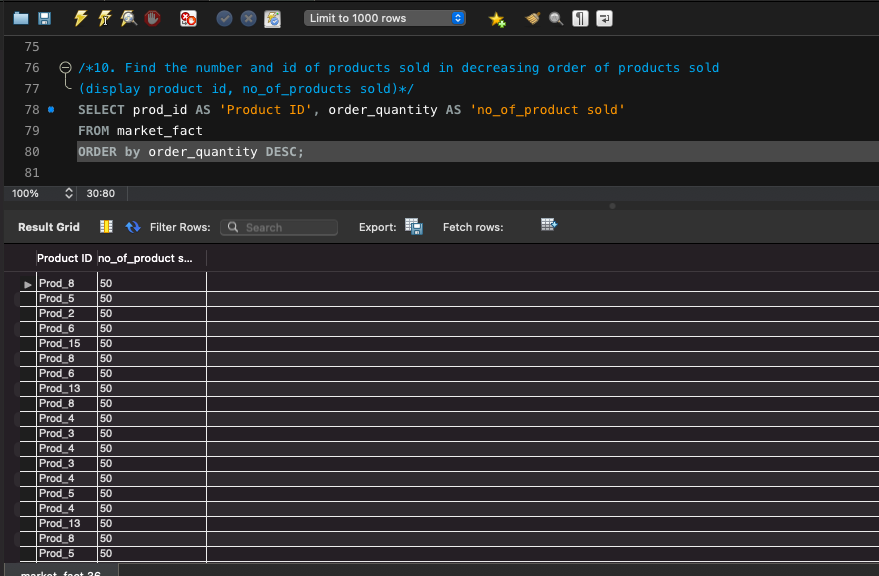
**/\*10. Find the number and id of products sold in decreasing order of products sold**

**(display product id, no\_of\_products sold)\*/**

SELECT prod\_id AS 'Product ID', order\_quantity AS 'no\_of\_product sold'

FROM market\_fact

ORDER by order\_quantity DESC;



**/\*11. Display product Id and product sub category whose produt category belongs to**

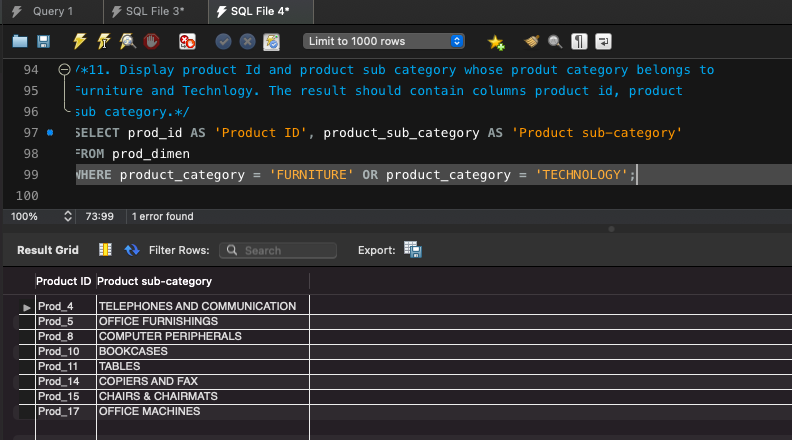
**Furniture and Technlogy. The result should contain columns product id, product**

**sub category.\*/**

SELECT prod\_id AS 'Product ID', product\_sub\_category AS 'Product sub-category'

FROM prod\_dimen

WHERE product\_category = 'FURNITURE' OR product\_category = 'TECHNOLOGY';



**/\*12. Display the product categories in descending order of profits (display the product**

**category wise profits i.e. product\_category, profits)?\*/**

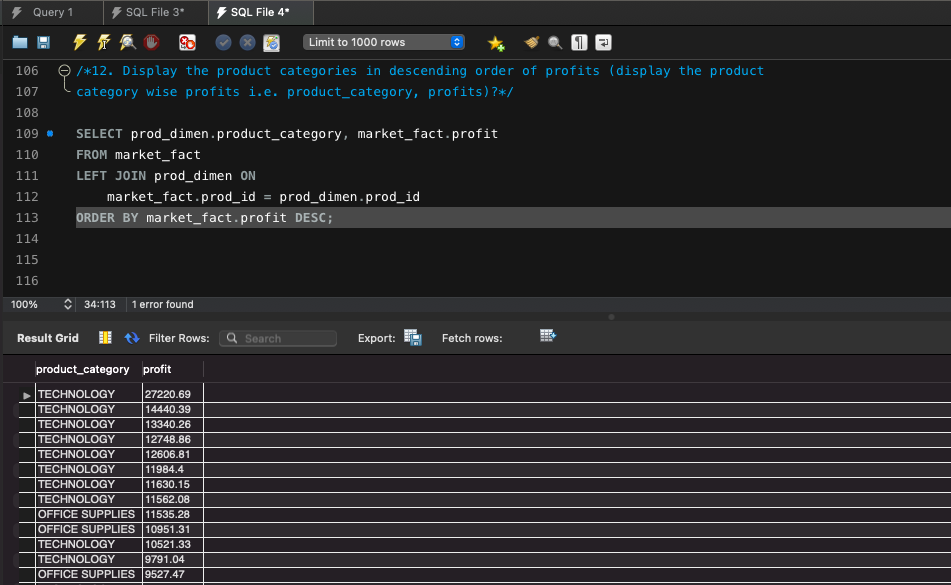
SELECT prod\_dimen.product\_category, market\_fact.profit

FROM market\_fact

LEFT JOIN prod\_dimen ON

market\_fact.prod\_id = prod\_dimen.prod\_id

ORDER BY market\_fact.profit DESC;



**/\*13. Display the product category, product sub-category and the profit within each**

**subcategory in three columns. \*/**

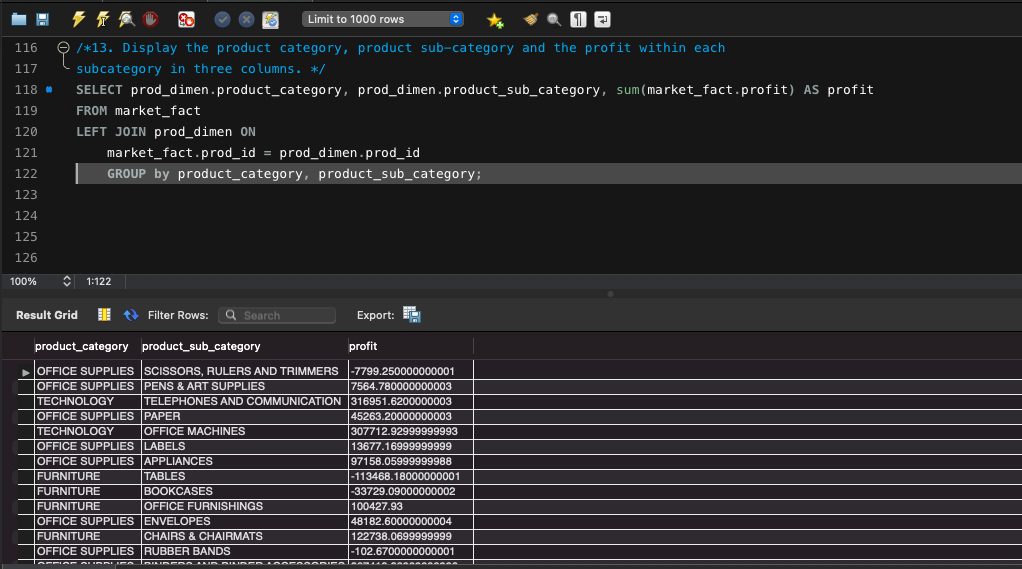
SELECT prod\_dimen.product\_category, prod\_dimen.product\_sub\_category, sum(market\_fact.profit) AS profit

FROM market\_fact

LEFT JOIN prod\_dimen ON

market\_fact.prod\_id = prod\_dimen.prod\_id

GROUP by product\_category, product\_sub\_category;



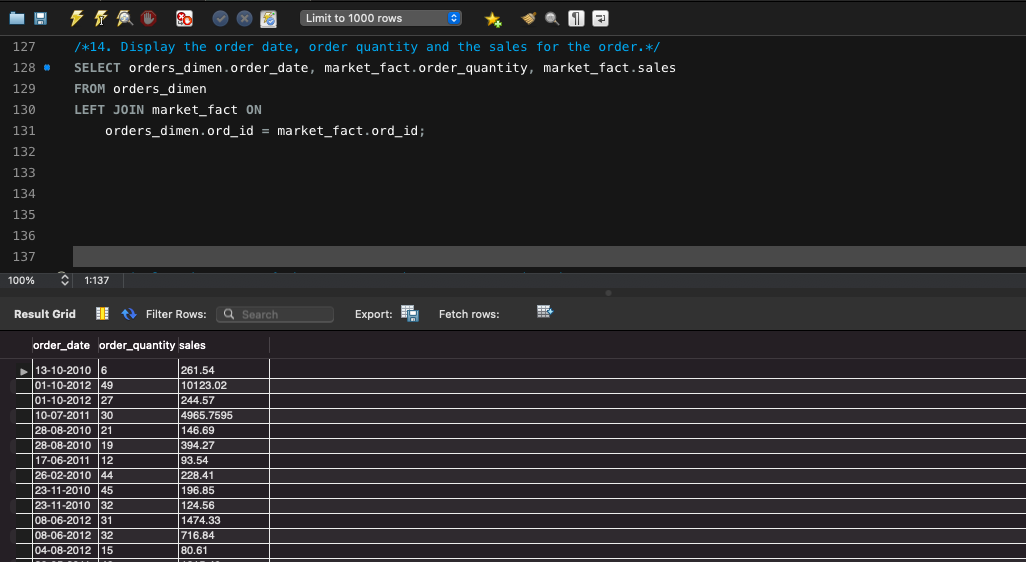
**/\*14. Display the order date, order quantity and the sales for the order.\*/**

SELECT orders\_dimen.order\_date, market\_fact.order\_quantity, market\_fact.sales

FROM orders\_dimen

LEFT JOIN market\_fact ON

orders\_dimen.ord\_id = market\_fact.ord\_id;



**/\*15. Display the names of the customers whose name contains the**

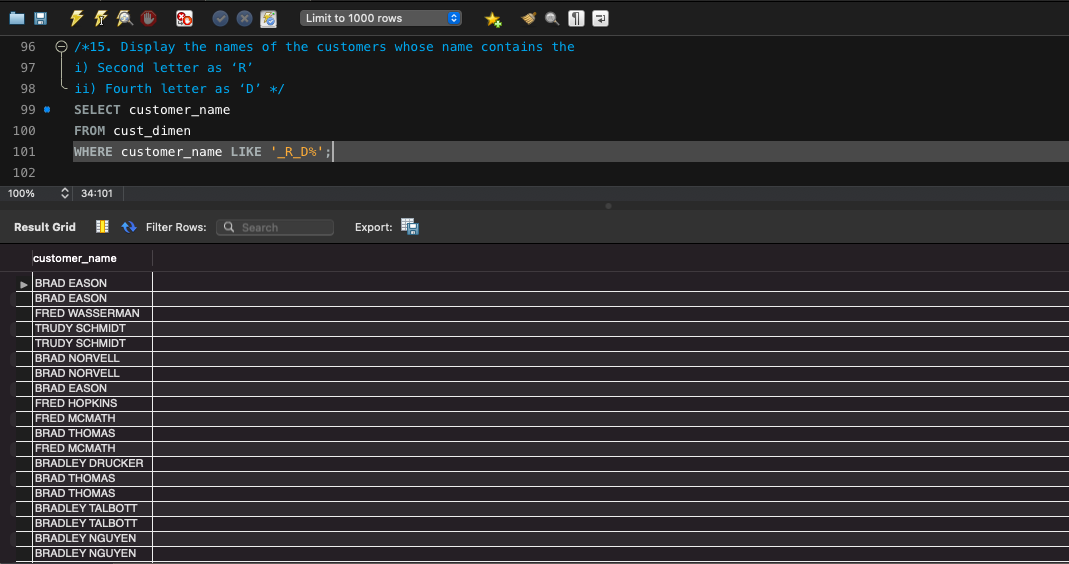
**i) Second letter as ‘R’**

**ii) Fourth letter as ‘D’ \*/**

SELECT customer\_name

FROM cust\_dimen

WHERE customer\_name LIKE '\_R\_D%'



**/\*16. Write a SQL query to make a list with Cust\_Id, Sales, Customer Name and**

**their region where sales are between 1000 and 5000.\*/**

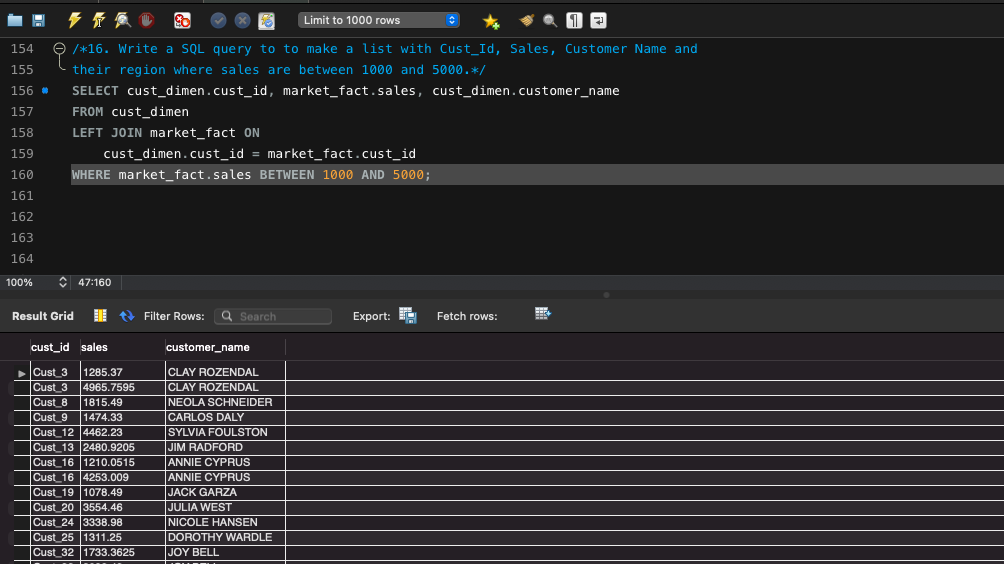
SELECT cust\_dimen.cust\_id, market\_fact.sales, cust\_dimen.customer\_name

FROM cust\_dimen

LEFT JOIN market\_fact ON

cust\_dimen.cust\_id = market\_fact.cust\_id

WHERE market\_fact.sales BETWEEN 1000 AND 5000;



**/\*17. Write a SQL query to find the 3rd highest sales.\*/**

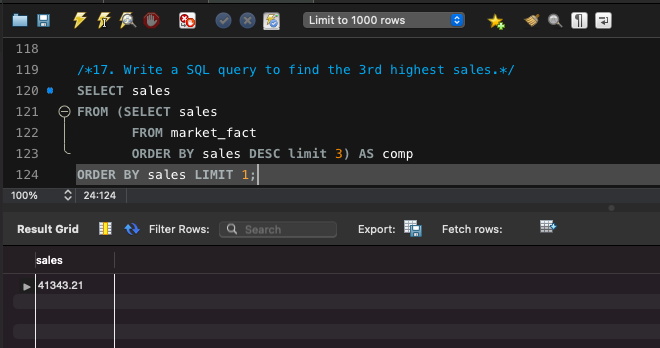
SELECT sales

FROM (SELECT sales

FROM market\_fact

ORDER BY sales DESC limit 3) AS comp

ORDER BY sales LIMIT 1;



**/\*18. Where is the least profitable product-subcategory shipped the most? For the least**

**profitable product sub-category, display the region-wise no\_of\_shipments and the**

**profit made in each region in decreasing order of profits (i.e. region, no\_of\_shipments, profit\_in\_each\_region)**

**→Note: You can hardcode the name of the least profitable product subcategory \*/**

SELECT c.Region as "Region",count(m.Ship\_id) as "No of Shipments",

round(sum(m.Profit),2) as "Profit in each region"

FROM market\_fact m

join cust\_dimen c on m.Cust\_id = c.Cust\_id

join prod\_dimen p on m.Prod\_id = p.Prod\_id

WHERE Product\_Sub\_Category = (

SELECT p.Product\_Sub\_Category

FROM market\_fact m

join prod\_dimen p on m.Prod\_id = p.Prod\_id

GROUP BY Product\_Sub\_Category

ORDER BY sum(m.Profit)

LIMIT 1)

GROUP BY c.Region

ORDER BY sum(m.Profit);

