DATA DESCRIPTION FOR SUPERSTORE DATA:

Note: Here I have given VARCHAR() instead of TEXT and INT to pass the values without errors(we can Give VARCHAR() to any PRIMARY DATA for length varying purpose as length of character)

Superstore_data types in my own words.

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 (VARCHAR()): 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 (VARCHAR()): Unique Order ID

4. prod_dimen: Details of product category and sub category

    Product_Category (TEXT): Product Category

Product_Sub_Category (VARCHAR()): Product Sub Category

Prod_id (VARCHAR()): 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 (VARCHAR()): Unique Shipment ID

Primary Keys and Foreign Keys for this dataset

1. cust_dimen

   Primary Key: Cust_id

   Foreign Key: NA

2. market_fact

   Primary Key: NA

   Foreign Key: Ord_id, Prod_id, Ship_id, Cust_id

3. orders_dimen

   Primary Key: Ord_id

   Foreign Key: NA

4. prod_dimen

   Primary Key: Prod_id, Product_Sub_Category

   Foreign Key: NA

5. shipping_dimen

   Primary Key: Ship_id

   Foreign Key: NA

SUPER_STORE DATABASE QUERIES:

```
SELECT customer_Name  "customer_Name",customer_segment "customer_segment"
FROM cust_dimen;

SELECT * FROM cust_dimen
ORDER BY Customer_Name DESC;

SELECT order_ID, order_Date, ord_id
FROM orders_dimen
WHERE Order_Priority='HIGH';

SELECT
SUM(sales) AS total_sales, AVG(sales) AS avg_sales
FROM market_fact;

SELECT MAX(sales), MIN(sales)
FROM market_fact;

SELECT region, COUNT(*) AS no_of_customers
FROM cust_dimen
GROUP BY region
ORDER BY no_of_customers DESC;

SELECT region, COUNT(*) AS no_of_customers
FROM cust_dimen
GROUP BY region
HAVING
no_of_customers >= ALL (          SELECT COUNT(*) AS no_of_customers
FROM cust_dimen
GROUP BY region );
```

```
SELECT c.customer_name, COUNT(*) AS no_of_tables_purchased
FROM market_fact m
INNER JOIN cust_dimen c ON m.cust_id = c.cust_id
WHERE c.region = 'atlantic'
AND m.prod_id = ( SELECT prod_id
FROM prod_dimen
WHERE product_sub_category = 'tables')
GROUP BY m.cust_id, c.customer_name;

SELECT Customer_Name AS "Customer Name" ,Customer_Segment AS "no. of small businees owners"
FROM cust_dimen WHERE Province="ONTARIO" AND Customer_Segment="SMALL BUSINESS";

SELECT prod_id AS product_id, COUNT(*) AS no_of_products_sold
FROM market_fact
GROUP BY prod_id
ORDER BY no_of_products_sold DESC;

SELECT Prod_id , Product_Sub_Category
FROM prod_dimen
WHERE Product_Category IN("TECHNOLOGY" AND "FURNITURE");

SELECT p.Product_Category AS "Product Category" , round(sum(m.Profit), 2) AS "Profits"
FROM market_fact m
JOIN Prod_dimen p ON m.Prod_id=p.Prod_id
GROUP BY p.Product_Category
ORDER BY sum(m.Profit) DESC;

SELECT p.product_category, p.product_sub_category, SUM(m.profit) AS profits
FROM market_fact m
INNER JOIN prod_dimen p ON m.prod_id = p.prod_id
GROUP BY p.product_category, p.product_sub_category;
```

```
SELECT Customer_Name
FROM cust_dimen
WHERE Customer_Name LIKE'_R%';

SELECT Customer_Name
FROM cust_dimen
WHERE Customer_Name LIKE'___D%';

SELECT  a.Cust_id,a.customer_Name,a.Region,
b.sales
FROM cust_dimen a,market_fact b
WHERE a.cust_id=b.cust_id
AND b.sales BETWEEN 1000 AND 5000;

SELECT sales
FROM
(SELECT sales
FROM market_fact
ORDER BY sales DESC
LIMIT 1) AS Comp
ORDER BY sales
LIMIT 3;

SELECT c.region, COUNT(distinct s.ship_id) AS no_of_shipments, SUM(m.profit) AS
profit_in_each_region
FROM market_fact m
INNER JOIN cust_dimen c ON m.cust_id = c.cust_id
INNER JOIN shipping_dimen s ON m.ship_id = s.ship_id
INNER JOIN prod_dimen p ON m.prod_id = p.prod_id
WHERE p.product_sub_category IN
(SELECT p.product_sub_category
FROM market_fact m
INNER JOIN
prod_dimen p ON m.prod_id = p.prod_id
```

```
GROUP BY p.product_sub_category
HAVING SUM(m.profit) <= ALL
(SELECT SUM(m.profit) AS profits
FROM market_fact m
INNER JOIN prod_dimen p ON m.prod_id = p.prod_id
GROUP BY p.product_sub_category))
GROUP BY c.region
ORDER BY profit_in_each_region DESC;
```