***Task 1: Understanding the data in hand***

1. **Describe the data in hand in your own words.**

🡪The dataset was provided in “csv” format thus it’s cleaned before importing into SQL workbench.

🡪In the given Database of Superstore, in all there are 5 tables.

*Cust\_dimen, market\_fact, orders\_dimen, prod\_dimen and shipping\_ dimen.*

*Cust\_dimen*: table provides the overall information about the customers.

*Prod\_dimen*: table provides the information about the categories and sub-categories of products.

*Shipping\_dimen*: table provides the shipping information about the orders.

*Orders\_dimen*: table provides the dates of orders received with their priorities.

*Market\_fact*: shows the detail of sales along with profit, shipping cost, discount and product base margin.

1. **Identify and list the Primary Keys and Foreign Keys for this dataset provided to you.**

**🡪**

|  |  |  |
| --- | --- | --- |
| Table Name | Primary Key | Foreign Key |
| Cust\_dimen | **Cust\_id** | **NA** |
| Market\_fact | **NA** | **Cust\_id, Ord\_id, Prod\_id, Ship\_id** |
| Orders\_dimen | **Ord\_id** | **NA** |
| Prod\_dimen | **Prod\_id** | **NA** |
| Shipping\_dimen | **Ship\_id** | **Order\_id** |

***Task 2: Basic and advance 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 Segment' 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 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, 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 market\_fact table.*

🡪SELECT max(sales) as 'maximum sales', min(sales) as 'minimum 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 region, COUNT(\*) AS no\_of\_customers FROM cust\_dimen GROUP BY region ORDER BY no\_of\_customers DESC;

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

🡪 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 );

*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(\*) 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;

*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(\*) as no\_of\_small\_business\_owner from cust\_dimen where province='ontario' and customer\_segment='small business';

*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, COUNT(\*) AS no\_of\_products\_sold

FROM

market\_fact

GROUP BY prod\_id

ORDER BY no\_of\_products\_sold DESC;

*11. Display product Id and product sub category whose product category belongs to Furniture and Technology. The result should contain columns product id, product sub category.*

🡪SELECT prod\_id, 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

p.product\_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

ORDER BY profits DESC;

*13. Display the product category, product sub-category and the profit within each subcategory in three columns.*

🡪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;

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

🡪select a.order\_date, b.order\_quantity ,b.sales from orders\_dimen a INNER JOIN market\_fact b ON a.ord\_id=b.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 b.cust\_id, b.customer\_name, b.region, a.sales

FROM cust\_dimen b

INNER JOIN market\_fact a

WHERE b.cust\_id=a.cust\_id

AND a.sales BETWEEN 1000 AND 5000;

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

🡪 SELECT sales FROM market\_fact ORDER BY sales DESC

LIMIT 2, 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, 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;