SQL CaseStudy- 1

Problem Statement:

You are a database administrator. You want to use the data to answer afewquestions about your customers, especially about the sales and profit comingfrom different states, money spent in marketing and various other factorssuchasCOGS (Cost of Goods Sold), budget profit etc. You plan on using theseinsightsto help find out which items are being sold the most. You have beenprovidedwith the sample of the overall customer data due to privacy issues. But youhopethat these samples are enough for you to write fully functioning SQLqueriestohelp answer the questions. Dataset: The 3 key datasets for this case study:

a. FactTable: The Fact Table has 14 columns mentioned belowand4200rows. Date, ProductID, Profit, Sales, Margin, COGS, Total Expenses, Marketing, Inventory, Budget Profit, Budget COGS, Budget Margin, Budget Sales, and Area Code
   Note: COGS stands for Cost of Goods Sold

b. ProductTable: The ProductTable has four columns named Product Type, Product, ProductID, and Type. It has 13 rows which can be brokendowninto further details to retrieve the information mentioned in theFactTable.

c. LocationTable: Finally, the LocationTable has 156 rows andfollowsasimilar approach to ProductTable. It has four columns named AreaCode, State, Market, and Market Size.

Tasks to be performed:

1. Display the number of states present in the LocationTable.

2. How many products are of regular type?

3. How much spending has been done on marketing of product ID 1?

4. What is the minimum sales of a product?

5. Display the max Cost of Good Sold (COGS).

6. Display the details of the product where product type is coffee.

7. Display the details where total expenses are greater than 40.

8. What is the average sales in area code 719?

9. Find out the total profit generated by Colorado state.

10. Display the average inventory for each product ID.

11. Display state in a sequential order in a Location Table.

12. Display the average budget of the Product where the average budget margin should be greater than 100.

13. What is the total sales done on date 2010-01-01?

14. Display the average total expense of each product ID on an individual date.

15. Display the table with the following attributes such as date, productID, product_type, product, sales, profit, state, area_code.

16. Display the rank without any gap to show the sales wise rank.

17. Find the state wise profit and sales.

18. Find the state wise profit and sales along with the productname.

19. If there is an increase in sales of 5%, calculate the increasedsales.

20. Find the maximum profit along with the product ID and producttype.

21. Create a stored procedure to fetch the result according to the product typefrom Product Table.
22. Write a query by creating a condition in which if the total expenses is lessthan60 then it is a profit or else loss.

23. Give the total weekly sales value with the date and product IDdetails. Useroll-up to pull the data in hierarchical order.

24. Apply union and intersection operator on the tables which consist of attribute area code.

25. Create a user-defined function for the product table to fetch a particular product type based upon the user's preference.

26. Change the product type from coffee to tea where product IDis 1 andundoit.

27. Display the date, product ID and sales where total expenses are between 100 to 200.

28. Delete the records in the Product Table for regular type.

29. Display the ASCII value of the fifth character from the columnProduct

```
--1. Display the number of states present in the LocationTable.
select count(distinct State) from LocationTable

--2. How many products are of regular type?
select count(*) from productTable group by Type having type='regular'

--3. How much spending has been done on marketing of product ID 1?
select sum(total_expenses),ProductID from factTable where ProductID=1

--4. What is the minimum sales of a product?
select * from factTable where sales=(select min(sales) from factTable)

--5. Display the max Cost of Good Sold (COGS).
select max(cogs) from factTable

--6. Display the details of the product where product type is coffee.
select * from productTable where Product_Type='coffee'

--7. Display the details where total expenses are greater than 40.
select * from  factTable where Total_Expenses>40

 --8. What is the average sales in area code 719?
 select avg(sales) as avg_sales_in_719 from factTable where Area_Code=719

 --9. Find out the total profit generated by Colorado state.
 select sum(profit)  as total_profit
 from factTable F inner join LocationTable L
```

```sql
 on  F.Area_Code=L.AreaCode
 where state='colorodo'

--10. Display the average inventory for each product ID.
SELECT ProductID, AVG(CAST(Inventory AS DECIMAL)) AS avg_inventory
FROM factTable
GROUP BY ProductID;

--11. Display state in a sequential order in a Location Table.
select state from LocationTable order by state

--12. Display the average budget of the Product where the average budget margin should
be greater than 100
SELECT AVG(Budget_Margin) AS avg_budget
FROM factTable
GROUP BY ProductID
HAVING AVG(Budget_Margin) > 100;

--13. What is the total sales done on date 2010-01-01?
select sum(sales) from factTable where date='2010-01-01'

--14. Display the average total expense of each product ID on an individual date.
select date, ProductID, avg(Total_Expenses) as avg_total_exp from factTable group by
Date, ProductID

--15. Display the table with the following attributes such as date, productID,
product_type, product, sales, profit, state, area_code.
select Date, F.ProductID, product_type, product, sales, profit, state, area_code
from factTable F
inner join LocationTable L on F.Area_Code=L.AreaCode
inner join productTable P on F.ProductID=P.ProductID

--16.Display the rank without any gap to show the sales wise rank.
SELECT
DENSE_RANK() OVER (ORDER BY sales DESC) AS sales_rank,
date, productid
FROM
facttable

--17. Find the state wise profit and sales.
select state, sum(profit) as total_profit,sum(sales) as total_sales
from facttable F inner join LocationTable L on F.area_code =L.areacode
group by state


--18. Find the state wise profit and sales along with the productname.
select P.Product, sum(f.sales), sum(f.Profit), L.State
from factTable F
inner join LocationTable L on F.Area_Code=L.AreaCode
inner join productTable P on F.ProductID=P.ProductID
group by State, P.Product;

--19. If there is an increase in sales of 5%, calculate the increasedsales.
SELECT ProductID,
Sales AS OriginalSales,
Sales * 0.05 AS IncreasedSales,
Sales + (Sales * 0.05) AS TotalSalesWithIncrease
FROM FactTable;
```

```sql
--20.Find the maximum profit along with the product ID and producttype
SELECT f.ProductID, p.Product_Type, MAX(f.Profit) AS MaximumProfit
FROM FactTable f
JOIN ProductTable p ON f.ProductID = p.ProductID
GROUP BY f.ProductID, p.Product_Type;

--21.Create a stored procedure to fetch the result according to the product typefrom
Product Table

CREATE PROCEDURE GetProductsByType
@Product_Type VARCHAR(255)
AS
BEGIN

SELECT ProductID, Product, Product_Type, Type
FROM ProductTable
WHERE Product_Type = @Product_type;

END;

--22.Write a query by creating a condition in which if the total expenses is lessthan
60 then it is a profit or else loss.
SELECT *,
CASE
WHEN Total_Expenses < 60 THEN 'Profit'
ELSE 'Loss'
END AS ProfitOrLoss
FROM FactTable;

--23.. Give the total weekly sales value with the date and product IDdetails. Useroll-
up to pull the data in hierarchical order.
SELECT
DATEPART(WEEK, Date) AS WeekNumber,
Date,
ProductID,
SUM(Sales) AS WeeklySales
FROM
FactTable
GROUP BY
ROLLUP((DATEPART(WEEK, Date), Date), ProductID);

--24.Apply union and intersection operator on the tables which consist of attribute
area code.
SELECT AreaCode FROM LocationTable
UNION
SELECT Area_Code FROM factTable;

SELECT AreaCode FROM LocationTable
intersect
SELECT Area_Code FROM factTable
```

```
--25.Create a user-defined function for the product table to fetch a particular
product type based upon the user’s preference.

CREATE FUNCTION GetProductsByType_UDF (
    @ProductType VARCHAR(255)  -- Parameter to filter products by type
)
RETURNS TABLE
AS
RETURN
(
SELECT ProductID, Product, Product_Type, Type
FROM productTable
WHERE Product_Type = @ProductType
);

-- 26. Change the product type from coffee to tea where product ID is 1 and undo it.
BEGIN TRANSACTION;

UPDATE ProductTable
SET Product_Type = 'tea'
WHERE ProductID = 1;

COMMIT TRANSACTION;
         -- Undo the change (rollback the transaction)
ROLLBACK TRANSACTION;

--27. Display the date, product ID and sales where total expenses are between 100 to
200.
SELECT Date, ProductID, Sales
FROM FactTable
WHERE Total_Expenses BETWEEN 100 AND 200;

--28. Delete the records in the Product Table for regular type.
DELETE FROM ProductTable
WHERE Type = 'regular';

--29.Display the ASCII value of the fifth character from the columnProduct
SELECT ASCII(SUBSTRING(Product, 5, 1)) AS ASCII_Value
FROM ProductTable;
```