1.Displaying the data

SELECT * FROM final

LIMIT 5;

Output:

ID	CustomerID	Gender	Location	Tenure_Months	Transaction_ID	Transaction_Date	Product_SK
0	17850	M	Chicago	12	16679	1/1/2019	GGOENEBJ0
1	17850	M	Chicago	12	16680	1/1/2019	GGOENEBJ0
2	17850	M	Chicago	12	16696	1/1/2019	GGOENEBQ0
3	17850	M	Chicago	12	16699	1/1/2019	GGOENEBQ0
4	17850	M	Chicago	12	16700	1/1/2019	GGOENEBJ0

2. Use SELECT, WHERE, ORDER BY, GROUP BY

a) SELECT CustomerID, Gender, Location, Online_Spend

FROM final;

CustomerID	Gender	Location	Online_Spend
15100	M	California	2424.5
14688	F	New York	2424.5
14688	F	New York	2424.5
14688	F	New York	2424.5
17850	M	Chicago	1576.38
17850	M	Chicago	1576.38
17850	M	Chicago	1576.38

b) SELECT CustomerID, Gender, Location, Tenure_Months

FROM final

WHERE Gender = 'F' AND Location = 'Chicago';

CustomerID	Gender	Location	Tenure_Months
17511	F	Chicago	25
17511	F	Chicago	25
17511	F	Chicago	25
17511	F	Chicago	25
17511	F	Chicago	25
13705	F	Chicago	39
13705	F	Chicago	39
13705	F	Chicago	39
13705	F	Chicago	39
13705	F	Chicago	39
13747	F	Chicago	21

c) SELECT CustomerID, Online_Spend, Offline_Spend

FROM final

ORDER BY Online_Spend DESC;

CustomerID	Online_Spend	Offline_Spend
15144	4556.93	3000
14467	4556.93	3000
17621	4556.93	3000
14688	4556.93	3000
14467	4556.93	3000
17894	4556.93	3000
14776	4556.93	3000
14527	4556.93	3000
15144	4556.93	3000
14776	4556.93	3000
17894	4556.93	3000

d) SELECT Location, COUNT(*) AS Total_Customers, AVG(Online_Spend) AS Avg_Online_Spend

FROM final

GROUP BY Location;

Location	Total_Customers	Avg_Online_Spend
Chicago	18240	1878.8350076754507
California	16008	1885.9289767616012
New York	11097	1935.0218608633081
New Jersey	4470	1912.2813042505343
Washington DC	2709	1849.7919822813178

3.SubQueries

a) SELECT *

FROM final

WHERE Online_Spend > (

SELECT AVG(Online_Spend) FROM final

);

ID	CustomerID	Gender	Location	Tenure_Months	Transaction_ID	Transaction_Date	Product_SKI
22	17850	M	Chicago	12	16731	1/1/2019	GGOENEBQ0
23	13047	M	California	43	16684	1/1/2019	GGOENEBQ0
24	13047	M	California	43	16684	1/1/2019	GGOENEBQ0
25	13047	M	California	43	16688	1/1/2019	GGOENEBB0
26	13047	M	California	43	16689	1/1/2019	GGOENEBJ0
27	12583	M	Chicago	33	16694	1/1/2019	GGOENEBB0
28	15100	M	California	49	16712	1/1/2019	GGOENEBJ0
29	14688	F	New York	46	16732	1/1/2019	GGOENEBJ0
30	14688	F	New York	46	16733	1/1/2019	GGOENEBB0
31	14688	F	New York	46	16736	1/1/2019	GGOENEBJ0

b) FROM (

SELECT Location, AVG(Online_Spend) AS AvgSpend

FROM final

GROUP BY Location

) AS avg_data

ORDER BY AvgSpend DESC;

Location	AvgSpend
New York	1935.0218608633081
New Jersey	1912.2813042505343
California	1885.9289767616012
Chicago	1878.8350076754507
Washington DC	1849.7919822813178

c) SELECT *

FROM final

WHERE CustomerID IN (

SELECT CustomerID

FROM final

WHERE Online_Spend > 500

);

ID	CustomerID	Gender	Location	Tenure_Months	Transaction_ID	Transaction_Date	Product_SK
12	17850	M	Chicago	12	16720	1/1/2019	GGOENEBJ(
13	17850	M	Chicago	12	16720	1/1/2019	GGOENEBQ(
14	17850	M	Chicago	12	16721	1/1/2019	GGOENEBJ(
15	17850	M	Chicago	12	16722	1/1/2019	GGOENEBJ(
16	17850	M	Chicago	12	16722	1/1/2019	GGOENEBQ(
17	17850	M	Chicago	12	16723	1/1/2019	GGOENEBB(
18	17850	M	Chicago	12	16723	1/1/2019	GGOENEBQ(
19	17850	M	Chicago	12	16724	1/1/2019	GGOENEBB(
20	17850	M	Chicago	12	16725	1/1/2019	GGOENEBB(
21	17850	M	Chicago	12	16727	1/1/2019	GGOENEBB(

d) SELECT *

FROM final t1

WHERE Online_Spend > (

SELECT AVG(Online_Spend)

FROM final t2

WHERE t2.Location = t1.Location

);

ID	CustomerID	Gender	Location	Tenure_Months	Transaction_ID	Transaction_Date	Product_SKI
0	17850	M	Chicago	12	16679	1/1/2019	GGOENEBJ0
1	17850	M	Chicago	12	16680	1/1/2019	GGOENEBJ0
2	17850	M	Chicago	12	16696	1/1/2019	GGOENEBQ0
3	17850	M	Chicago	12	16699	1/1/2019	GGOENEBQ0
4	17850	M	Chicago	12	16700	1/1/2019	GGOENEBJ0
5	17850	M	Chicago	12	16701	1/1/2019	GGOENEBJ0
6	17850	M	Chicago	12	16702	1/1/2019	GGOENEBJ0
7	17850	M	Chicago	12	16703	1/1/2019	GGOENEBQ0
8	17850	M	Chicago	12	16704	1/1/2019	GGOENEBJ0
9	17850	M	Chicago	12	16710	1/1/2019	GGOENEBJ0

4.Use aggregate functions (SUM, AVG)

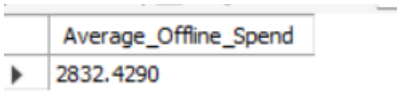
a) SELECT SUM(Online_Spend) AS Total_Online_Spend

FROM final;

Total_Online_Spend
99491823.09999622

b) SELECT AVG(Offline_Spend) AS Average_Offline_Spend

FROM final;



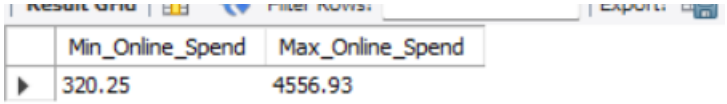
Average_Offline_Spend
2832.4290

c) SELECT

MIN(Online_Spend) AS Min_Online_Spend,

MAX(Online_Spend) AS Max_Online_Spend

FROM final;



Min_Online_Spend	Max_Online_Spend
320.25	4556.93

5.Creating view for analysis

a) CREATE VIEW customer_spending_summary AS

SELECT

CustomerID,

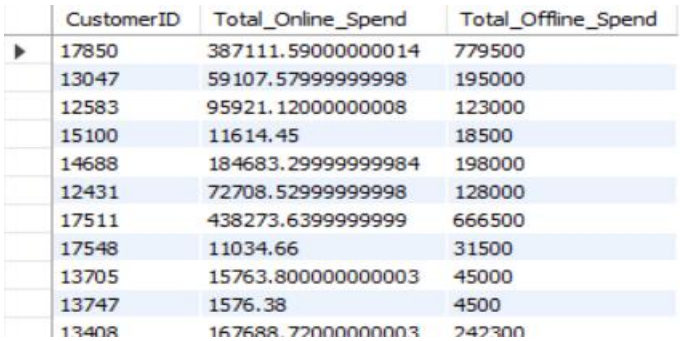
SUM(Online_Spend) AS Total_Online_Spend,

SUM(Offline_Spend) AS Total_Offline_Spend

FROM final

GROUP BY CustomerID;

SELECT * FROM customer_spending_summary;



CustomerID	Total_Online_Spend	Total_Offline_Spend
17850	387111.59000000014	779500
13047	59107.57999999998	195000
12583	95921.12000000008	123000
15100	11614.45	18500
14688	184683.29999999984	198000
12431	72708.52999999998	128000
17511	438273.6399999999	666500
17548	11034.66	31500
13705	15763.800000000003	45000
13747	1576.38	4500
13408	167688.72000000003	242300

b) CREATE VIEW monthly_discount_analysis AS

SELECT

Month,

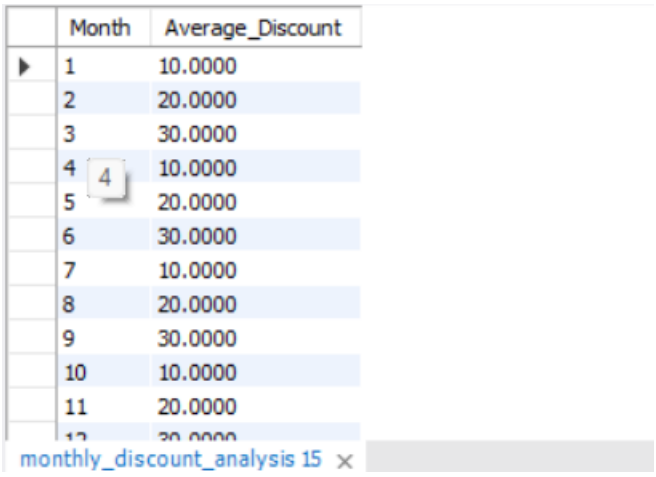
AVG(Discount_pct) AS Average_Discount

FROM final

GROUP BY Month

ORDER BY Month;

SELECT * FROM monthly_discount_analysis;



Month	Average_Discount
1	10.0000
2	20.0000
3	30.0000
4	10.0000
5	20.0000
6	30.0000
7	10.0000
8	20.0000
9	30.0000
10	10.0000
11	20.0000

monthly_discount_analysis 15

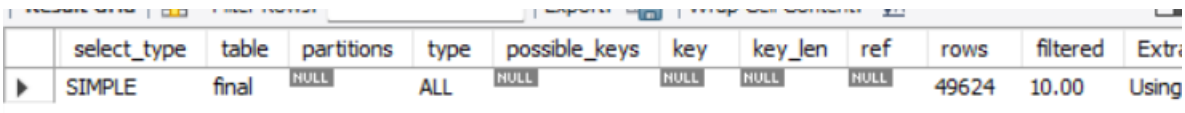
6. **Optimize queries with indexes**

CREATE INDEX idx_customer_id ON final(CustomerID);

CREATE INDEX idx_online_spend ON final(Online_Spend);

CREATE INDEX idx_month_discount ON final(Month, Discount_pct);

EXPLAIN SELECT * FROM final WHERE Location = 'Delhi';



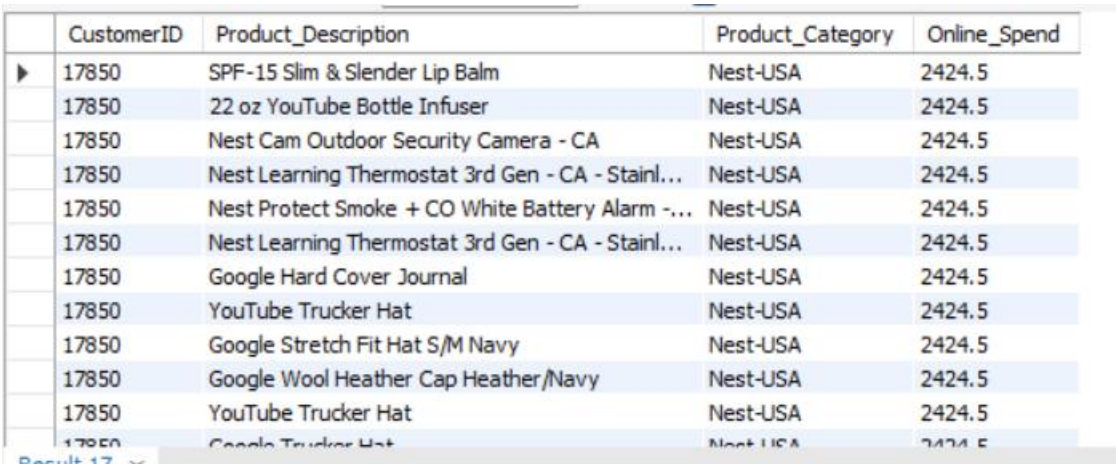
select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
SIMPLE	final	NULL	ALL	NULL	NULL	NULL	NULL	49624	10.00	Using

7. Using Joins (Left, Right, Inner)

a) SELECT t.CustomerID, c.Product_Description, t.Product_Category, t.Online_Spend

FROM final t

LEFT JOIN task04 c ON t.CustomerID = c.CustomerID;



CustomerID	Product_Description	Product_Category	Online_Spend
17850	SPF-15 Slim & Slender Lip Balm	Nest-USA	2424.5
17850	22 oz YouTube Bottle Infuser	Nest-USA	2424.5
17850	Nest Cam Outdoor Security Camera - CA	Nest-USA	2424.5
17850	Nest Learning Thermostat 3rd Gen - CA - Stainl...	Nest-USA	2424.5
17850	Nest Protect Smoke + CO White Battery Alarm -...	Nest-USA	2424.5
17850	Nest Learning Thermostat 3rd Gen - CA - Stainl...	Nest-USA	2424.5
17850	Google Hard Cover Journal	Nest-USA	2424.5
17850	YouTube Trucker Hat	Nest-USA	2424.5
17850	Google Stretch Fit Hat S/M Navy	Nest-USA	2424.5
17850	Google Wool Heather Cap Heather/Navy	Nest-USA	2424.5
17850	YouTube Trucker Hat	Nest-USA	2424.5

b) SELECT c.CustomerID, c.Tenure_Months, t.Product_Category, t.Online_Spend

FROM final t

RIGHT JOIN task04 c ON t.CustomerID = c.CustomerID;

CustomerID	Tenure_Months	Product_Category	Online_Spend
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5
17850	12	Nest-USA	2424.5

C) SELECT c.CustomerID, t.Product_Category, t.Online_Spend, t.Month

FROM final t

INNER JOIN task04 c ON t.CustomerID = c.CustomerID;

CustomerID	Product_Category	Online_Spend	Month
17850	Notebooks & Jour...	417.73	1
17850	Nest-Canada	1312.5	1
17850	Nest-Canada	1312.5	1
17850	Nest-Canada	1312.5	1
17850	Nest-Canada	1049.73	1
17850	Bottles	937.58	1
17850	Housewares	1312.5	1
17850	Nest-USA	2424.5	1
17850	Nest-USA	2424.5	1