SQL Assignment

Question 1. Write a query to return sales total grouped by channel id and product id. (No SQL extensions to group by required here). Sample of the output is below. What is the Explain plan cost and table access?

Description: Goal of this query is to retrieve and compile sales information from the "SALES Table". It specially determines the total sales for each Channel_id and Prod_id to provide the overall sales for each combination.

Aims and Objectives: Aims is to calculate the total sales amount for each unique combination of "Channel_id" and "Prod_id." This includes determining how much of each product was sold through each sales channel.

Objective is to compare the performance of different sales channels to understand which channels are selling the most, which are the least, and any variations in sales patterns.

Methods: In this we are using Sum aggregate function and Group By clause to get the total sales on Channel_id, Prod_id.

SQL Syntax Explanation:

1. From clause is used to retrieve the data from Sales table.
2. Then I selected Channel_id, Prod_id, Sum(amount_sold). (Were sum(amount_sold) is to show the aggregate Total_sales).
3. Grouping on channel_id, Prod_id.
   (With help of Group By clause Sales_Total is divided into groups on Channel-id, Prod_id).

Channel_id: This column represents the unique identifier for the sales channel, where the product was sold.
Prod_id: This column represents the unique identifier for the product being sold.
Total_Sales: This column represents the total sales amount for each combination of "channel_id" and "prod_id". It is calculated by adding the "amount_sold" values for each combination.

Results: In below Figure 1, we can see that the output I got are
Total Rows = 228, Time taken to execute = 0.134seconds.
Sum(amount_sold) is renamed as “Sales_Total” and products which are being sold through each sales channel.

Plan cost & Table access: In Figure 2,

- Time taken to execute = 0.127seconds.
- Cost = 47 → So it might be costly, in-terms of resource consumption.
- Cardinality = 204 → As it processed 204 rows.
- Cost = 17 → For a "TABLE ACCESS" operation, which implies that this specific table access operation is less resource-intensive as we have chosen smaller table.

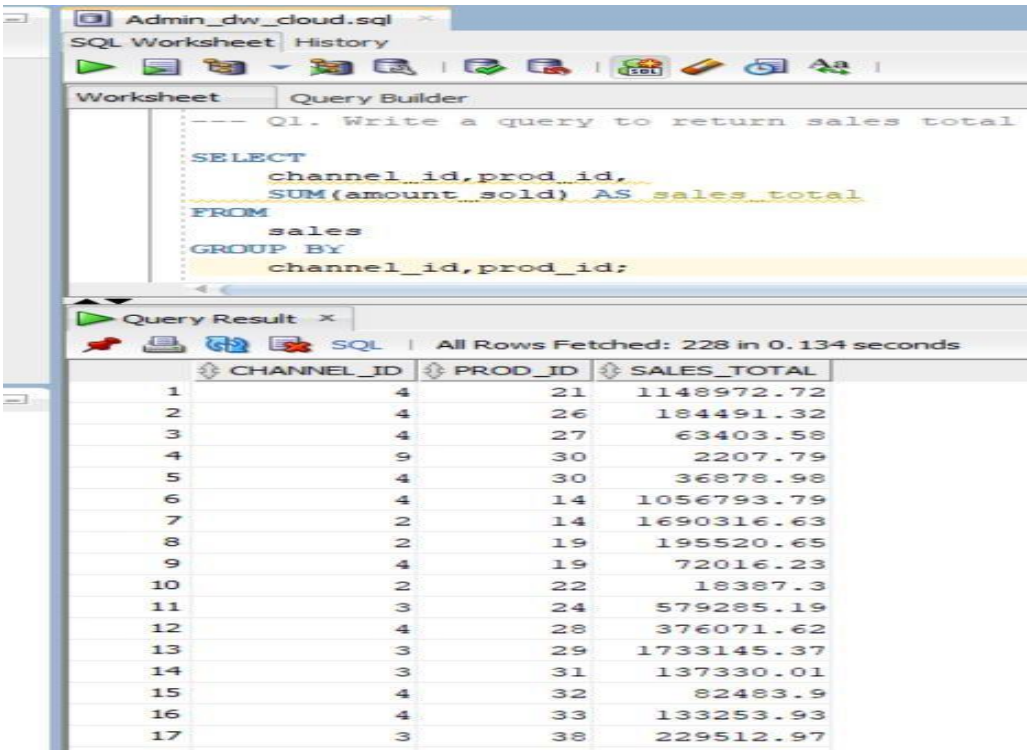


Figure 1 (Query 1 output)

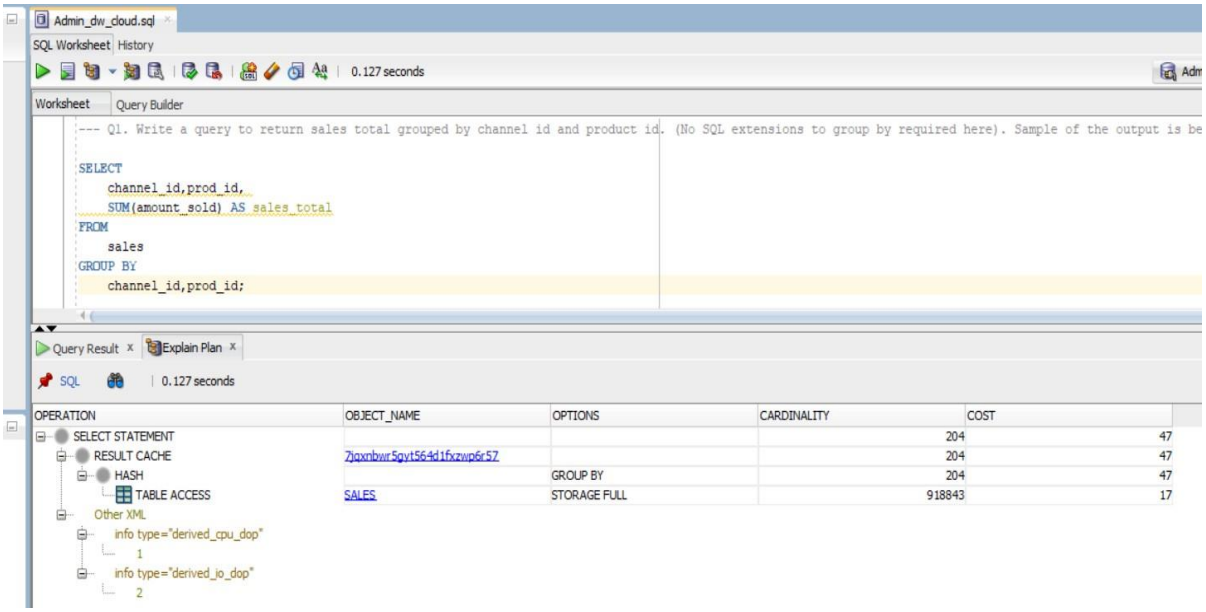


Figure 2 (Explain Cost)

Conclusions: This query helps us to identify which products are popular through which channels and which combinations might have higher or lower sales. Provides a structured summary of sales performance data in the above figure1.

Question 2. Create a Materialized View called SALES_CHAN_PROD_MV based on the query you wrote above for Question 1. Show the Materialized View being used by query rewrite by noting the Explain plan cost and table access. Ensure that you describe the output and reasons for the explain plan cost reduction.

Description: Goal is to create Materialized view (SALES_CHAN_PROD_MV) on Channel_id, Prod_id and sum(amount_sales), which can be computed automatically when the data is relevant and which results in faster query execution.

Aims and Objectives: Aim is to reduce resource consumption and improve data accessibility to enhance query performance determining how much of each product was sold through each sales channel.

Objective is used to check data integrity, query optimization, and decision-making support to compare the performance of different sales channels.

Methods:
Creating the materialized view to focus on defining the view's structure and
Refresh force on demand → (how data should be refreshed),
Enable Query rewrite → (by enabling query optimization),
Primary key → ensuring data integrity
On First question syntax in **(SQL Syntax Explanation)**

Results: In Figure 3, we can see that Materialized view is created and the output is executed/ Completed Task in 0.668 seconds.

Plan cost & Table access: In Figure 4, After Materialized View on Query 1 is created. We can see that "Cost is 2" which is very good in terms of resource consumption and Main goal in this query is to choose the plan that minimizes resource consumption while delivering the correct results.

It is a positive sign for query performance, as it suggests that the database optimizer has chosen an efficient plan to execute the query quickly.

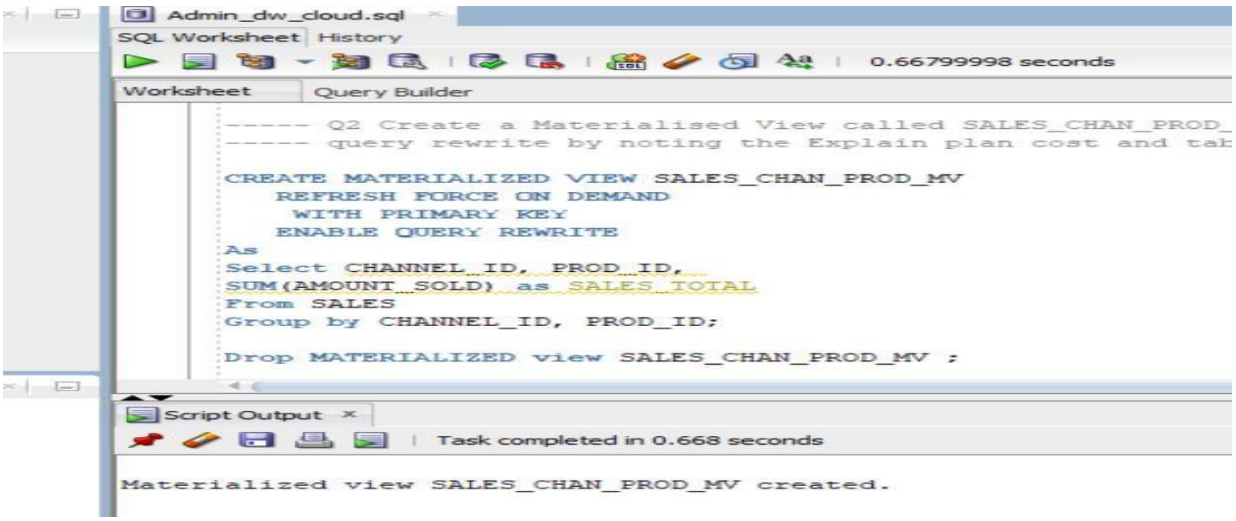


Figure 3 (Materialized view created)

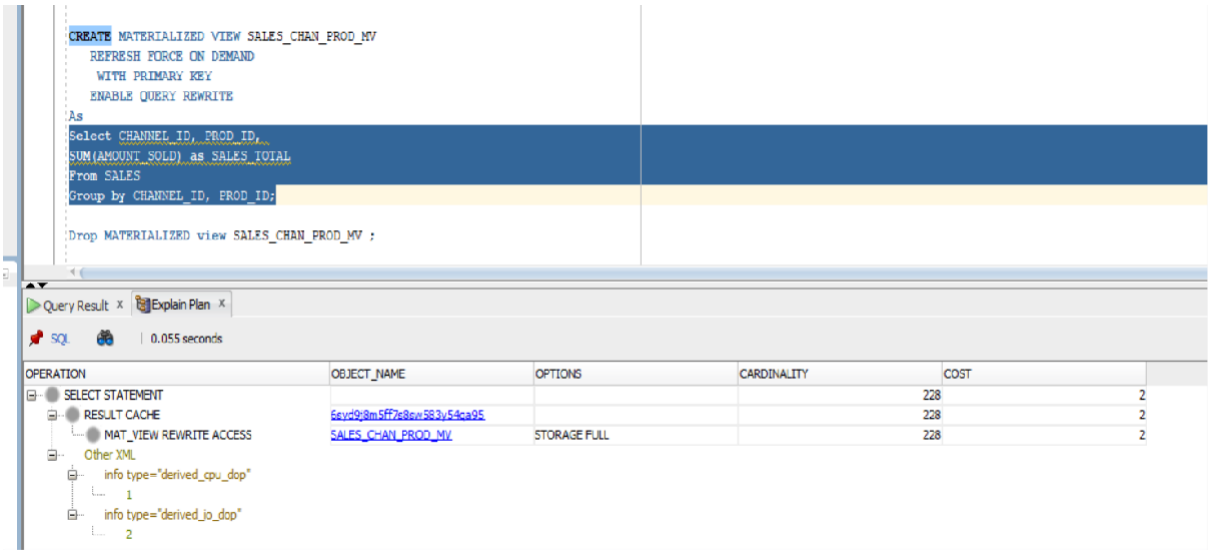


Figure 4 (Explain Cost)

Conclusions: Purpose of creating materialized view is to precomputed data which can improve query performance when analyzing sales data, as the aggregation has already been done and stored. So, Runtime/Executed time will be less and to minimizes resource consumption. With Materialized view usage cost is lowered that previous cost.

Questions 3. Write a query to return sales total grouped by channel description and product name. (No SQL extensions to GROUP BY required here). A sample of the output is as follows (225 rows). What is the Explain plan cost and table access?

Description: Goal is to retrieve, summarize sales data by joining Sales, Channels and Products tables and grouping them on basing Channel_desc, Prod_name. Making it easier to identify trends and patterns.

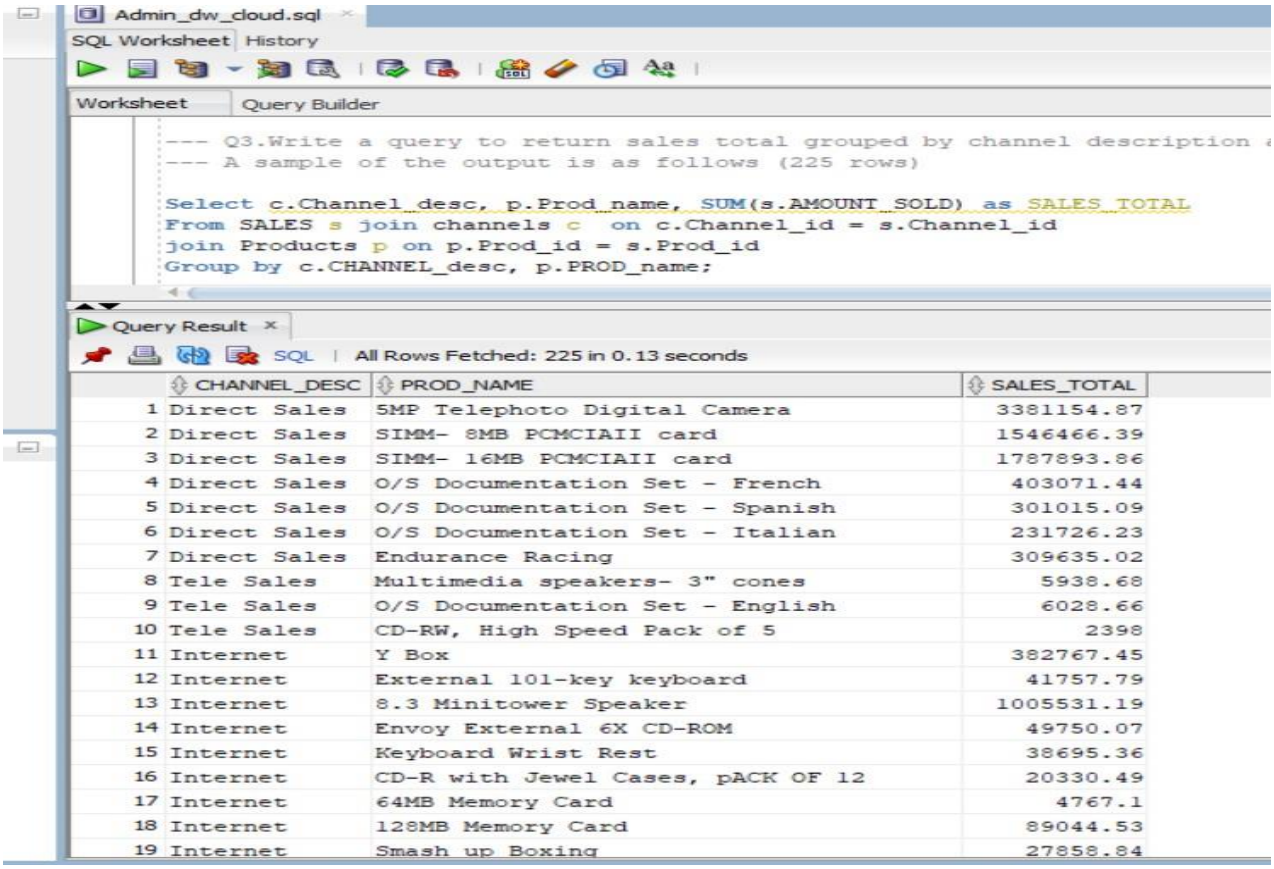
Aims and Objectives: Aim would be to focus on the total sales amounts of various products through different sales channels to check the trends and patterns.

Objective is to offer a structure view for better understanding the sales performance on each different channels and products.

Methods:

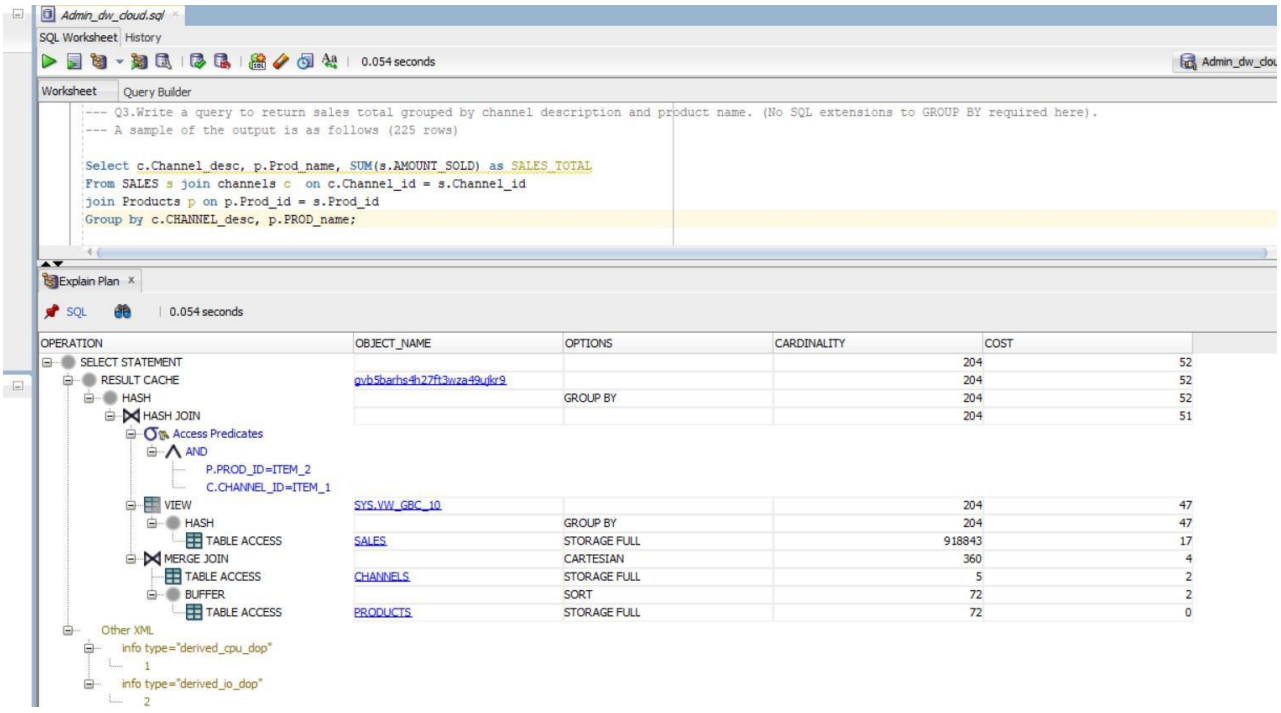
- This query is to retrieve data From sales table
- Then with help of join clause → Joining Sales, Channels & Products tables
- Grouping the result to provide Total_sales to each combination of Channel_desc and Prod_name.
- This will be helpful for analyzing sales performance and identifying which products are sold through which channels.

Results: In Figure 5, we can see Sales_Total output basing on each Prod_Name, Channel_Desc combination. **Plan cost & Table access:** In Figure 6, this we can see that "Cost & Table access value for Sales, Channels, Products.



	CHANNEL_DESC	PROD_NAME	SALES_TOTAL
1	Direct Sales	5MP Telephoto Digital Camera	3381154.87
2	Direct Sales	SIMM- 8MB PCMCIAII card	1546466.39
3	Direct Sales	SIMM- 16MB PCMCIAII card	1787893.86
4	Direct Sales	O/S Documentation Set - French	403071.44
5	Direct Sales	O/S Documentation Set - Spanish	301015.09
6	Direct Sales	O/S Documentation Set - Italian	231726.23
7	Direct Sales	Endurance Racing	309635.02
8	Tele Sales	Multimedia speakers- 3" cones	5938.68
9	Tele Sales	O/S Documentation Set - English	6028.66
10	Tele Sales	CD-RW, High Speed Pack of 5	2398
11	Internet	Y Box	382767.45
12	Internet	External 101-key keyboard	41757.79
13	Internet	8.3 Minitower Speaker	1005531.19
14	Internet	Envoy External 6X CD-ROM	49750.07
15	Internet	Keyboard Wrist Rest	38695.36
16	Internet	CD-R with Jewel Cases, pACK OF 12	20330.49
17	Internet	64MB Memory Card	4767.1
18	Internet	128MB Memory Card	89044.53
19	Internet	Smash up Boxing	27858.84

Figure 5 (Output 3)



OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
SELECT STATEMENT			204	52
RESULT CACHE	gvb5barhs4h27ft3wza49ujkr9		204	52
HASH		GROUP BY	204	52
HASH JOIN			204	51
Access Predicates				
AND				
P.PROD_ID=ITEM_2				
C.CHANNEL_ID=ITEM_1				
VIEW	SYS.VW_GBC_10		204	47
HASH		GROUP BY	204	47
TABLE ACCESS	SALES	STORAGE FULL	918843	17
MERGE JOIN		CARTESIAN	360	4
TABLE ACCESS	CHANNELS	STORAGE FULL	5	2
BUFFER		SORT	72	2
TABLE ACCESS	PRODUCTS	STORAGE FULL	72	0
Other XML				
info type="derived_cpu_dop"				
1				
info type="derived_io_dop"				
2				

Figure 6 (Explain Cost for Question 3)

Conclusions: It is used to analyze Sales data on each Channel and Product to give Sales strategies and insights on each product if it is successful or not in selling.

Question 4. Does the Query written in Question 3 above use the Materialized View SALES_CHAN_PROD_MV? If not explain why. Document your observations.

No, Query 3 is not using materialized view to fetch data from SALES_CHAN_PROD_MV as we can see the results are higher and not taking any advantage of materialized view. Where because of this query response times & computational would not be that faster.

In this cost = 47 is higher and table access for all the tables are (Sales = 17, Products = 2, Channels = 0) → Refer Figure 6.

Question 5. Create an appropriate Dimensional object to allow Q3 to use the Materialized View SALES_CHAN_PROD_MV.

Description: Goal is to create dimension object on query 3 using hierarchy, level, and attributes to determine on sales, products, and additional descriptive information at different levels. With help of user_dimensions running the syntax.

Aims and Objectives: Aim is to create structured and organized dimensional object for data and hierarchy representation on Sales, products, and categories.

Objective is to define level, create hierarchy and attribute to analyze the sales data for data analysis and reporting dimension.

Methods:

- Creating Dimension → Sales_Dimension
- Then adding levels and attributes and hierarchy
- Level → These levels are used to categorize and structure the data hierarchically, providing a framework for multidimensional analysis.
- Hierarchy → This hierarchy defines the relationships between the levels, indicating how they are organized.
- Join Key → This clause is used to specifies the relationship.
- Determine→ Determines various attributes related to individual.

Results: In Figure 7, We can see that dimension is created. → Sales_Dimension.
Execution time for dimension creation = 0.066 seconds.

In Figure 8, Dimension is executed by user_dimensions and were
Dimension name = Sales_Dimension
compilation status = “valid “
Invalid = N (No)
Revision = 1.
Execution time = 0.029 seconds with 1 Row.

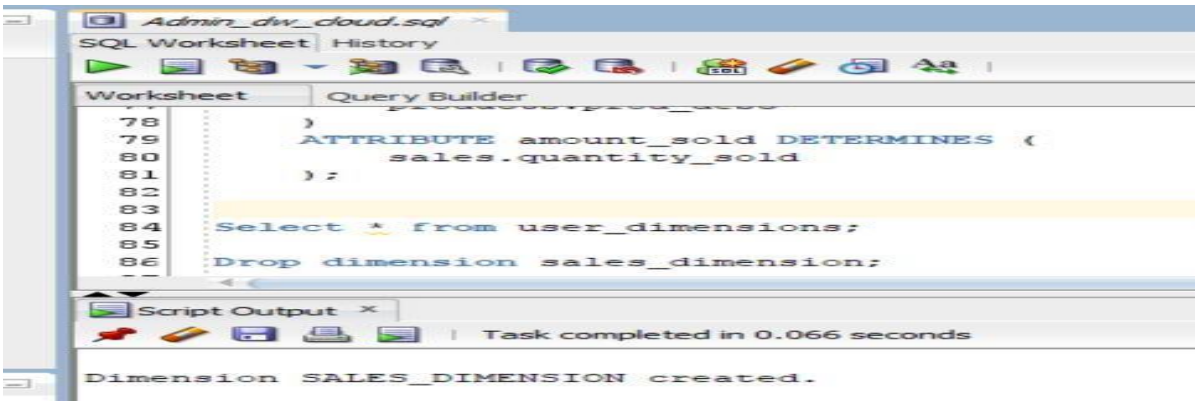


Figure 7 (Dimension Created)

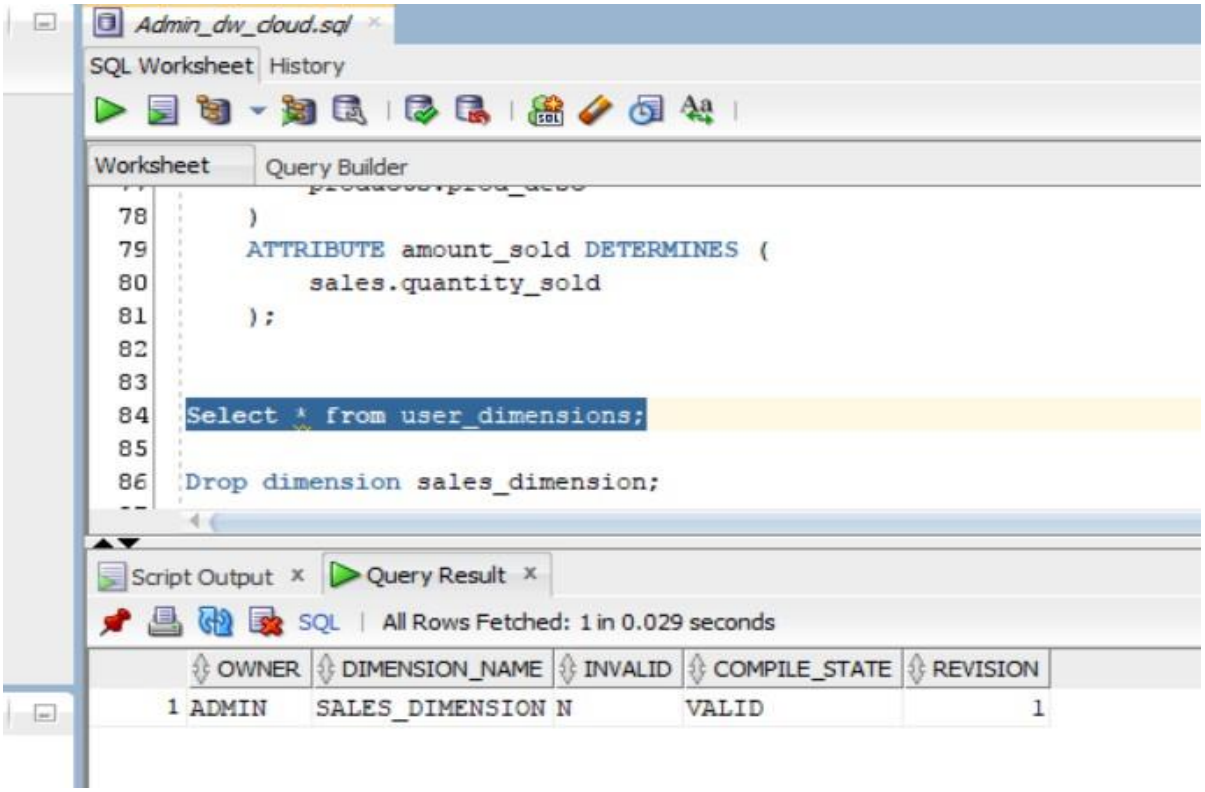


Figure 8 (Dimension result)

Conclusion: I have created dimensional object named "sales dimension" which uses join key and determines, attributes to aim structured and organized sales data for better analysis and reporting.

Question 6. Rerun Query 3. Does it use the materialized view SALES_CHAN_PROD_MV? What is the Explain plan cost and table access?

Yes, this query now uses materialized view SALES_CHAN_PROD_MV. In this cost and table has been reduced which is a positive. Where cost is 7 and Table access for (Channels = 2, Product = 0). Now runtime would be less and optimization would be reduced too. Figure 9.

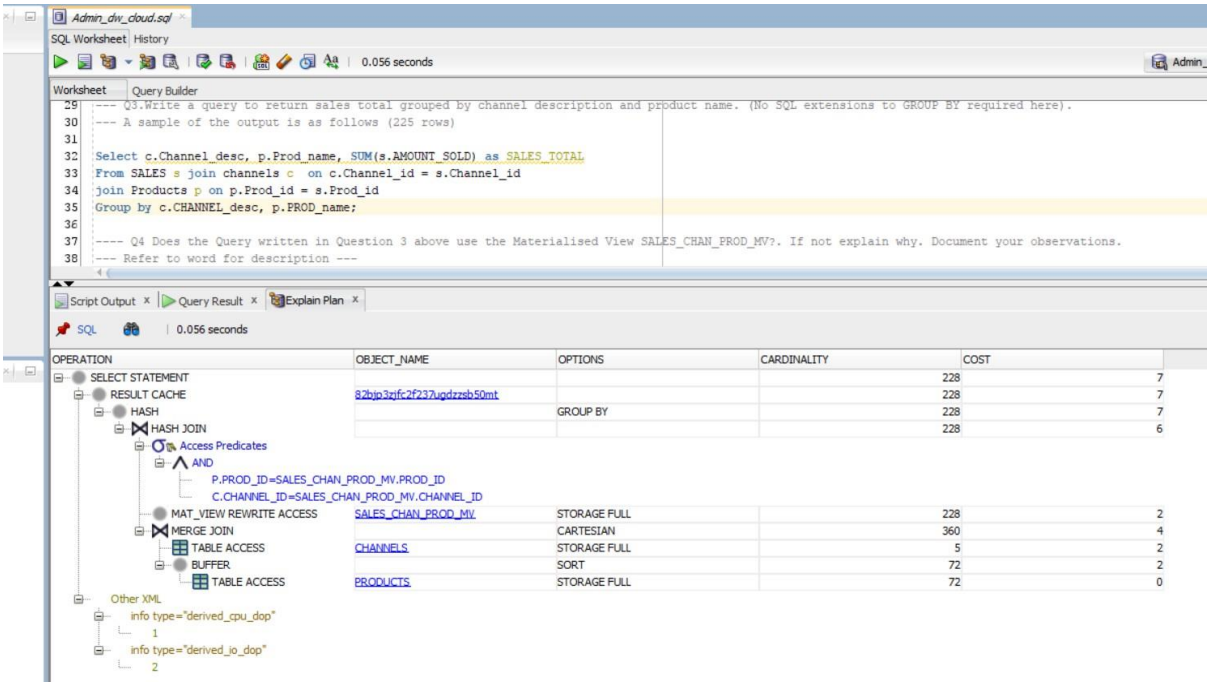


Figure 9 (Explain Cost and table access)

Question 7. Write a query to produce a report for management that require sales total grouped by channel description, product category and country name showing a total at each level of aggregation for France and Italy. Management do not want Peripherals and Accessories, Hardware or Photo product categories included in the report. Improve the readability of the report by using decode (29 records are returned as you see in the report in Figure 3).

Description: Goal of this query is to use the "DECODE" function for conditional labeling of columns, filtering the data for France and Italy while excluding specific products, categories. Using Roll up, grouping, grouping sets and cube.

Aims and Objectives: Aims and objectives are centered around multidimensional analysis, total sales calculation, data filtering, and the provision of valuable insights and reporting to support data-driven decision-making in a business or analytical context.

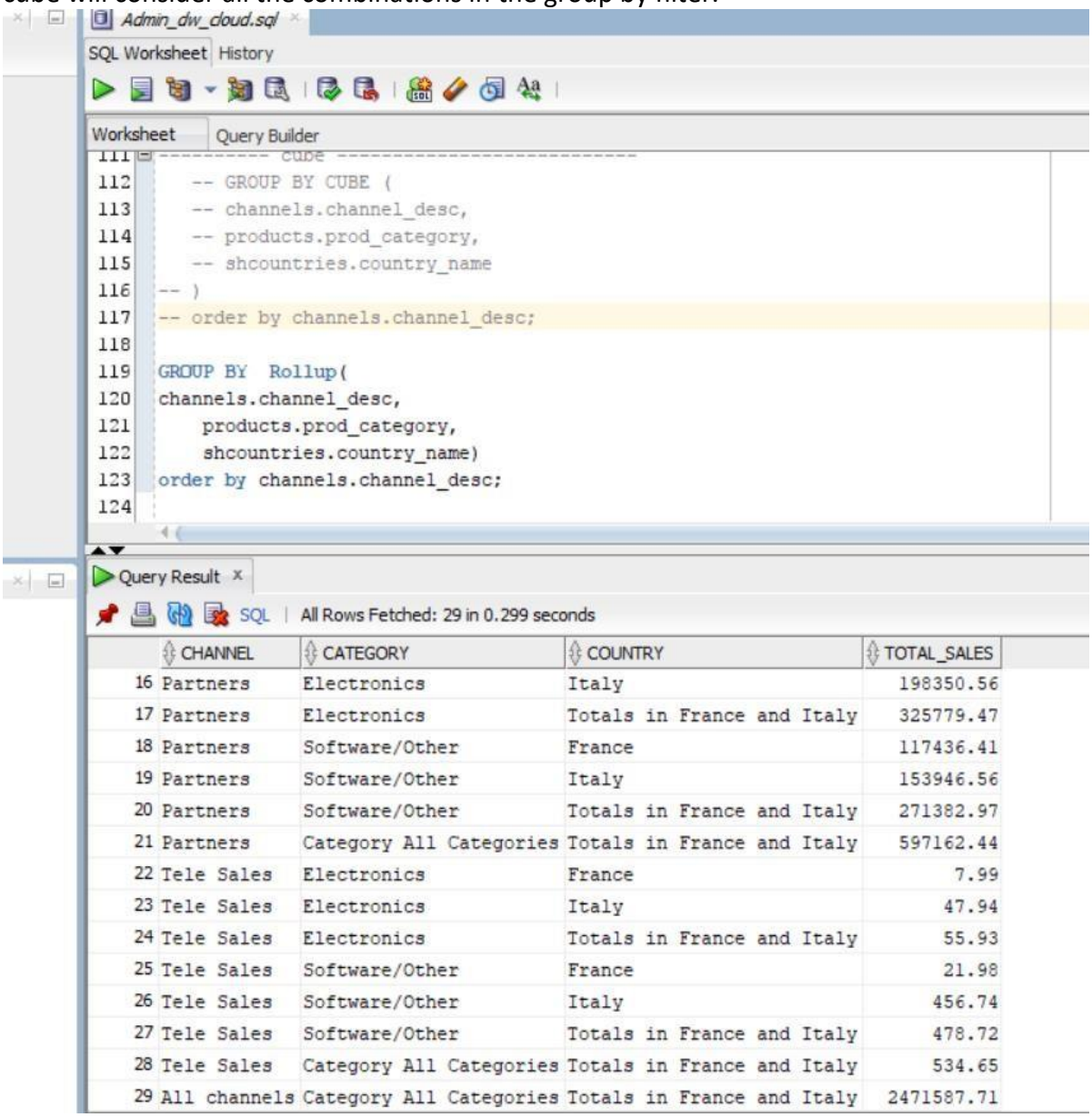
Method: I experimented on
Grouping sets, Rollup, Decode, Cube, grouping these are the methods used in this query

- Grouping is to get detailed level to a grand total.
- Cube is to get subtotals for every possible combination.
- Grouping sets is to create the desired aggregate levels.
- Rollup is to get reasonable number of subtotals

Result: In figure 10, we can see that Rows = 29, Total Sales = 2471587.71

Conclusion: The query uses rollup decode, grouping, cube and grouping sets results row = 29.

Rollup and grouping set function uses same output but cube has different output because cube will consider all the combinations in the group by filter.



	CHANNEL	CATEGORY	COUNTRY	TOTAL_SALES
16	Partners	Electronics	Italy	198350.56
17	Partners	Electronics	Totals in France and Italy	325779.47
18	Partners	Software/Other	France	117436.41
19	Partners	Software/Other	Italy	153946.56
20	Partners	Software/Other	Totals in France and Italy	271382.97
21	Partners	Category All Categories	Totals in France and Italy	597162.44
22	Tele Sales	Electronics	France	7.99
23	Tele Sales	Electronics	Italy	47.94
24	Tele Sales	Electronics	Totals in France and Italy	55.93
25	Tele Sales	Software/Other	France	21.98
26	Tele Sales	Software/Other	Italy	456.74
27	Tele Sales	Software/Other	Totals in France and Italy	478.72
28	Tele Sales	Category All Categories	Totals in France and Italy	534.65
29	All channels	Category All Categories	Totals in France and Italy	2471587.71

Figure 10