**ST1501 CA2 Group Task**

**Class: DAAA/FT/2A/01**

**Group No: 4**

**Group Members: Cody, Joaquin, Rachel, SongLing**

|  |  |
| --- | --- |
| **AdmNo** | **Name** |
| **2112617** | **Lim Ke Zhen Joaquin** |
| **2112688** | **Ho Ka Yee Rachel** |
| **2138000** | **Tan Shi Wei Cody** |
| **2112576** | **Li SongLing** |

**Group Task**

Data Warehouse Schema

|  |
| --- |
|  |

Queries explanation

|  |  |  |
| --- | --- | --- |
| Query | Explanation | SQL script |
| 1 | The query shows the product with the highest sales generated to the lowest with its profits, total quantity sold, and the average discount given to customers.  The team can infer from the query that despite the high list price and slightly lower average discount as compared to other products, the ’Trek Slash 8 27.5 – 2016’ still brings in the highest revenue amongst all products. | SELECT f.product\_id, p.product\_name AS 'Product Name', p.list\_price AS 'List Price', ROUND(SUM(f.sales), 2) AS 'Total Sales', ROUND(SUM(f.profit), 2) AS 'Total Profit', SUM(f.quantity) AS 'Quantity Sold', CONCAT(CAST(AVG(f.discount)\*100 AS INT),'%') AS 'Average Discount'  FROM factTable f, productDim p  WHERE f.product\_id = p.product\_id  GROUP BY f.product\_id, p.product\_name, p.list\_price  ORDER BY [Total Profit] DESC; |
| 2 | The query shows the staffs with the highest revenue generated in descending order, along with their store code.  From this, the team can infer which staffs should be rewarded based on the revenue they bring in.  We can also infer in general, ST2 staff bring in the most revenue, followed by ST1 and finally ST3. | SELECT (s.first\_name + ' '+ s.last\_name) 'Staff Name', ROUND(SUM(f.sales \* f.discount), 2) 'Total Sales', f.store\_id 'Store Code'  FROM factTable f, staffDim s  WHERE f.staff\_id = s.staff\_id  GROUP BY (s.first\_name + ' '+ s.last\_name), f.store\_id  ORDER BY [Total Sales] DESC; |
| 3 | The query shows the total sales generated weekly in descending order of the total sales.  From this query, the team can infer that the sales have been generally increasing each year. However, the sales within each year fluctuate; across all the years, there is no specific season where the sales are the highest. | SELECT SUM(f.sales) AS 'Total Sales',t.Year, t.WeekOfYear AS 'Week Of Year', t.MonthName AS 'Month'  FROM factTable f, timeDim t  WHERE f.time\_id = t.time\_id  GROUP BY t.Year, t.WeekOfYear, t.MonthName  ORDER BY [Total Sales] DESC; |
| 4 | The query shows the customer with the highest sales generated in descending order of total sales, state and city.  From this query, the team can identify which states and/or cities to open their next store at. We can also infer in general, that the state of New York generates the most amount of sales. However, the sales within each state fluctuate and there is no specific city where the sales are the highest. | SELECT (c.first\_name + ' '+ c.last\_name) 'Customer Name', ROUND(SUM(f.sales \* f.discount), 2) 'Total Sales', COUNT(f.order\_id) 'Quantity of Items Bought', c.[state] 'State', c.city 'City'  FROM factTable f, customerDim c  WHERE f.customer\_id = c.customer\_id  GROUP BY (c.first\_name + ' '+ c.last\_name), c.[state], c.city  ORDER BY [Total Sales] DESC, c.[state], c.city; |
| 5 | The query shows the top 3 products sold in each category, along with the respective brand, quantity sold, remaining stock, and how much profit the product makes.  The team can infer from here which category is selling best, and can be observed here is Mountain Bikes, followed by Electric Bikes, and Cyclocross Bikes.  There is also a trend here that Trek Brand sells better in general compared to other brands. From this query, every Trek Bike has sales of about an average of $250K, followed by Surly with an average of $230K and Electra with $60K. However, Trek and Electra both surpass Surly in popularity. | SELECT \* FROM  (SELECT c.category\_name AS 'Category Name', b.brand\_name AS 'Brand Name', p.product\_name AS 'Product Name', SUM(f.quantity) AS 'Quantity Sold', p.stock AS Stock, ROUND(SUM(f.sales), 2) AS 'Total Sales', ROUND(SUM(f.profit), 2) AS 'Total Profit', RANK() OVER (PARTITION BY c.category\_name ORDER BY ROUND(SUM(f.sales), 2) DESC) AS [Ranking]  FROM factTable f, categoryDim c, brandDim b, productDim p  WHERE f.product\_id = p.product\_id AND p.category\_id = c.category\_id AND p.brand\_id = b.brand\_id  GROUP BY c.category\_name, b.brand\_name, p.product\_name, p.stock) AS ranktable  WHERE ranking IN (1,2,3)  ORDER BY [Category Name] ,[Total Sales] DESC; |