**Felipe Lima - 109290055**

USE `HW\_5\_DW`;

-- 1. Show a list of Customer Name, Gender, Sales Person’s Name and Sales Person's City for all products sold on September 2015, whose Sales Price is more than 20 and Quantity sold is more than 8.

SELECT C.CustomerName, C.Gender, SP.SalesPersonName, SP.City

FROM Dim\_Customer C

INNER JOIN Fact\_ProductSales FPS ON C.CustomerID = FPS.CustomerID

INNER JOIN Dim\_SalesPerson SP ON FPS.SalesPersonID = SP.SalesPersonID

INNER JOIN Dim\_Date D ON FPS.SalesDateKey = D.DateKey

WHERE D.MONTHNAME = "September" AND D.YEAR = "2015" AND FPS.SalesPrice > 20 AND FPS.Quantity > 8

GROUP BY C.CustomerName, C.Gender, SP.SalesPersonName, SP.City;

A close up of a sign

Description automatically generated

-- 2. Show a list of Store Name, Store's City and Product Name for all products sold on March 2017, whose Product Cost is less than 50 and store located in 'Boulder'.

SELECT S.StoreName, S.City, P.ProductName

FROM Dim\_Store S

INNER JOIN Fact\_ProductSales FPS ON S.StoreID = FPS.StoreID

INNER JOIN Dim\_Product P ON P.ProductKey = FPS.ProductID

INNER JOIN Dim\_Date D ON FPS.SalesDateKey = D.DateKey

WHERE D.MONTHNAME = "March" AND D.YEAR = "2017" AND FPS.ProductCost < 50 AND S.City = "Boulder"

GROUP BY S.StoreName, S.City, P.ProductName;

Table

Description automatically generated

-- 3. Show a list of Top 2 Sales Person’s Name by their Total Revenue for 2017, i.e. Top 2 sales person with HIGHEST Total Revenue.

SELECT SP.SalesPersonName, SUM(FPS.SalesPrice \* FPS.Quantity) AS "Total Revenue"

FROM Dim\_SalesPerson SP

INNER JOIN Fact\_ProductSales FPS ON FPS.SalesPersonID = SP.SalesPersonID

INNER JOIN Dim\_Date D ON D.DateKey = FPS.SalesDateKey

WHERE D.YEAR = 2017

GROUP BY SP.SalesPersonName

ORDER BY SUM(FPS.SalesPrice \* FPS.Quantity) DESC

LIMIT 2;

Table

Description automatically generated

-- 4. Display a Customer Name and Total Revenue who has LOWEST Total Revenue in 2017.

SELECT C.CustomerName, SUM(FPS.SalesPrice \* FPS.Quantity) AS "Total Revenue"

FROM Dim\_Customer C

INNER JOIN Fact\_ProductSales FPS ON C.CustomerID = FPS.CustomerID

INNER JOIN Dim\_Date D ON D.DateKey = FPS.SalesDateKey

WHERE D.YEAR = 2017

GROUP BY C.CustomerName

ORDER BY SUM(FPS.SalesPrice \* FPS.Quantity) ASC

LIMIT 1;

A picture containing table

Description automatically generated

-- 5. Show a list of Store Name (in alphabetical order) and their 'Total Sales Price' for the year between 2010 and 2017.

SELECT S.StoreName, SUM(FPS.SalesPrice) AS "Total Sales Price"

FROM Dim\_Store S

INNER JOIN Fact\_ProductSales FPS ON S.StoreID = FPS.StoreID

INNER JOIN Dim\_Date D ON D.DateKey = FPS.SalesDateKey

WHERE D.YEAR >= 2010 AND D.YEAR <= 2017

GROUP BY S.StoreName

ORDER BY S.StoreName;

Table

Description automatically generated

-- 6. Display a list of Store Name, Product Name and their Total Profits from product name like 'Jasmine Rice' for 2010.

SELECT S.StoreName, P.ProductName, SUM((FPS.SalesPrice \* FPS.Quantity) - (FPS.ProductCost \* FPS.Quantity)) AS "Total Profit"

FROM Dim\_Store S

INNER JOIN Fact\_ProductSales FPS ON S.StoreID = FPS.StoreID

INNER JOIN Dim\_Product P ON P.ProductKey = FPS.ProductID

INNER JOIN Dim\_Date D ON D.DateKey = FPS.SalesDateKey

WHERE D.YEAR = 2010

GROUP BY S.StoreName, P.ProductName;

A picture containing sitting

Description automatically generated

-- 7. Display Total Revenue from 'ValueMart Boulder' Store for each Quarter during 2016, sort your result by Quarter in chronological order.

SELECT SUM(FPS.SalesPrice \* FPS.Quantity) AS "Total Revenue", D.QUARTER

FROM Fact\_ProductSales FPS

INNER JOIN Dim\_Store S ON S.StoreID = FPS.StoreID

INNER JOIN Dim\_Date D ON FPS.SalesDateKey = D.DateKey

WHERE S.StoreName = "ValueMart Boulder" AND D.YEAR = 2016

GROUP BY D.QUARTER

ORDER BY D.QUARTER;

Table

Description automatically generated

-- 8. Display Customer Name and Total Sales Price for all items purchased by customers Melinda Gates and Harrison Ford.

SELECT C.CustomerName, SUM(FPS.SalesPrice) AS "Total Sales Price"

FROM Dim\_Customer C

INNER JOIN Fact\_ProductSales FPS ON C.CustomerID = FPS.CustomerID

WHERE C.CustomerName = "Melinda Gates" OR C.CustomerName = "Harrison Ford"

GROUP BY C.CustomerName;

Table

Description automatically generated

-- 9. Display Store Name, Sales Price and Quantity for all items sold in March 12th 2017.

SELECT S.StoreName, FPS.SalesPrice, FPS.Quantity

FROM Dim\_Store S

INNER JOIN Fact\_ProductSales FPS ON S.StoreID = FPS.StoreID

INNER JOIN Dim\_Date D ON FPS.SalesDateKey = D.DateKey

WHERE D.DATE = '2017/03/12';

Table

Description automatically generated

-- 10. Display Sales Person’s Name and Total Revenue for the best performing Sales Person, i.e., the Sales Person with the HIGHEST Total Revenue.

SELECT SP.SalesPersonName, SUM(FPS.SalesPrice \* FPS.Quantity) AS "Total Revenue"

FROM Dim\_SalesPerson SP

INNER JOIN Fact\_ProductSales FPS ON SP.SalesPersonID = FPS.SalesPersonID

GROUP BY SP.SalesPersonName

ORDER BY SUM(FPS.SalesPrice \* FPS.Quantity) DESC

LIMIT 1;

A picture containing table

Description automatically generated

-- 11. Display the Top 3 Product Name by their HIGHEST Total Profit.

SELECT P.ProductName

FROM Dim\_Product P

INNER JOIN Fact\_ProductSales FPS ON P.ProductKey = FPS.ProductID

GROUP BY P.ProductName

ORDER BY SUM((FPS.SalesPrice \* FPS.Quantity) - (FPS.ProductCost \* FPS.Quantity)) DESC

LIMIT 3;

A picture containing text

Description automatically generated

-- 12. Display Year, MonthName and Total Revenue for the 1st 3 months (i.e. January, February and March) of 2017.

SELECT D.YEAR, D.MONTHNAME, SUM(FPS.SalesPrice \* FPS.Quantity) AS "Total Revenue"

FROM Dim\_Date D

INNER JOIN Fact\_ProductSales FPS ON D.DateKey = FPS.SalesDateKey

WHERE D.DATE >= '2017/01/01' AND D.DATE <= '2017/03/31'

GROUP BY D.YEAR, D.MONTHNAME;

Table

Description automatically generated

-- 13. Display Product Name, average product cost and average sales price for the products sold in 2017. Show averages rounded to 2 decimal places.

SELECT P.ProductName, ROUND(AVG(FPS.ProductCost),2) AS "Average Product Cost", ROUND(AVG(FPS.SalesPrice),2) AS "Average Sales Price"

FROM Dim\_Product P

INNER JOIN Fact\_ProductSales FPS ON P.ProductKey = FPS.ProductID

INNER JOIN Dim\_Date D ON FPS.SalesDateKey = D.DateKey

WHERE D.YEAR = 2017

GROUP BY P.ProductName;

Table

Description automatically generated

-- 14. Display Customer Name, average sales price and average quantity for all items purchased by customer Melinda Gates. Show averages rounded to 2 decimal places.

SELECT C.CustomerName, ROUND(AVG(FPS.SalesPrice),2) AS "Average Sales Price", ROUND(AVG(FPS.Quantity),2) AS "Average Quantity"

FROM Dim\_Customer C

INNER JOIN Fact\_ProductSales FPS ON C.CustomerID = FPS.CustomerID

WHERE C.CustomerName = "Melinda Gates"

GROUP BY C.CustomerName;

A picture containing graphical user interface

Description automatically generated

-- 15. Display Store Name, Maximum sales price and Minimum sales price for store located in 'Boulder' city. Show MIN / MAX rounded to 2 decimal places.

SELECT S.StoreName, ROUND(MAX(FPS.SalesPrice),2) AS "Maximum Sales Price" , ROUND(MIN(FPS.SalesPrice),2) AS "Minumum Sales Price"

FROM Dim\_Store S

INNER JOIN Fact\_ProductSales FPS ON S.StoreID = FPS.StoreID

WHERE S.City = "Boulder"

GROUP BY S.StoreName;

