**CIS 9440 Data Warehousing and Analytics Homework Assignment #1 Analytical SQL**

**A screenshot of a cell phone

Description automatically generated**

**4) Write a SQL statement to show the total dollar amount sold to all customers summarized by City of their headquarters and each month of each Year (YYYY-MM).**

SELECT c.customercity AS City\_of\_Headquarters,

TO\_CHAR(o.orderdate, 'YYYY-MM') AS Each\_Month\_of\_Each\_Year,

SUM(ol.orderedquantity \* ol.extendedprice) AS Total\_Dollars

FROM furniture.orders o

NATURAL JOIN furniture.orderline ol

NATURAL JOIN furniture.customer c

WHERE p.productfinish IN (‘Cherry’, ‘Oak’)

AND o.fulfillmentdate IS NOT NULL

GROUP BY c.customercity, TO\_CHAR(o.orderdate, 'YYYY-MM')

ORDER BY TO\_CHAR(o.orderdate, 'YYYY-MM');

**Results: full result 105**

|  |  |  |
| --- | --- | --- |
| **CITY\_OF\_HEADQUARTERS** | **EACH\_MONTH\_OF\_EACH\_YEAR** | **TOTAL\_DOLLARS** |
| Gainesville | 2014-11 | 400 |
| Carteret | 2014-12 | 6550.98 |
| Albany | 2015-01 | 9350.6 |
| Farmington | 2015-01 | 1625.5 |
| Boulder | 2015-02 | 9181 |
| Carteret | 2015-02 | 870 |
| Gainesville | 2015-02 | 1050 |
| Sacramento | 2015-02 | 4000 |
| Carteret | 2015-06 | 900 |
| Farmington | 2015-06 | 170 |
| Gainesville | 2015-06 | 175 |
| Las Cruces | 2015-06 | 293.4 |
| Plano | 2015-06 | 1592 |
| Rome | 2015-06 | 6702.4 |
| Sacramento | 2015-06 | 958 |
| Virginia Beach | 2015-06 | 5905 |
| Albany | 2015-07 | 341.96 |
| Boulder | 2015-07 | 1625 |
| Farmington | 2015-07 | 4196.25 |
| Las Cruces | 2015-07 | 685 |
| Oak Brook | 2015-07 | 750 |
| Sacramento | 2015-07 | 1650 |
| Gainesville | 2015-08 | 1734 |
| Rome | 2015-08 | 7675.59 |
| Las Cruces | 2015-09 | 576.8 |
| Syracuse | 2015-09 | 1149.6 |
| Farmington | 2015-12 | 4387 |
| Gainesville | 2015-12 | 422 |
| Albany | 2016-06 | 525 |
| Ft Walton Beach | 2016-06 | 3700 |

**5) Write a SQL statement to show a Ranking of total sales by Territory Name. The largest sales should be ranked #1.**

SELECT t.territoryname AS Territory\_Name,

SUM(ol.orderedquantity \* ol.extendedprice) AS Total\_Sales,

RANK() OVER (ORDER BY SUM(ol.orderedquantity \* ol.extendedprice) DESC) AS Ranking

FROM furniture.orders o

NATURAL JOIN furniture.orderline ol

NATURAL JOIN furniture.customer c

NATURAL JOIN furniture.customershipaddress cs

NATURAL JOIN furniture.territory t

WHERE o.shipadrsid = cs.shipaddressid

GROUP BY t.territoryname;

**Results:**

|  |  |  |
| --- | --- | --- |
| **TERRITORY\_NAME** | **TOTAL\_SALES** | **RANKING** |
| SouthWest | 160302.20 | 1 |
| SouthEast | 76566.40 | 2 |
| Central | 48082.00 | 3 |
| NorthWest | 36730.59 | 4 |
| Arizona | 30953.60 | 5 |
| NorthEast | 21237.56 | 6 |
| Colorado | 17975.98 | 7 |
| Alaska | 17849.00 | 8 |

**6) Write a SQL statement to show the product that has the largest percentage markup.**

SELECT p.productlineid, p.productdescription, p.productfinish, p.productstandardprice, p.productonhand, p.productcost,

(p.productstandardprice - p.productcost)/ p.productcost AS Percentage\_Markup

FROM furniture.product p

WHERE (p.productstandardprice - p.productcost)/ p.productcost =

(SELECT MAX((p.productstandardprice - p.productcost)/ p.productcost)

FROM furniture.product p);

**Results:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PRODUCTLINEID** | **PRODUCTDESCRIPTION** | **PRODUCTFINISH** | **PRODUCTSTANDARDPRICE** | **PRODUCTONHAND** | **PRODUCTCOST** | **PERCENTAGE\_MARKUP** |
| 4 | 6' Grandfather Clock | Oak | 890 | 0 | 231.4 | 2.84615384615384615384615384615384615385 |
| 2 | Amoire | Walnut | 1200 | 0 | 312 | 2.84615384615384615384615384615384615385 |

**7) Who is the “best” customer? Justify your rationale and back it up with queries and data. You may also wish to graph various data to support your justification. DO NOT just total up sales. Look at other tables and analysis as well (CLV and RFM for example).**

SELECT Customer\_Name, RFM\_Recency \* 100 + RFM\_Frequency \* 10 + RFM\_Monetary as RFM\_Combined

FROM (

SELECT Customer\_Name,

ntile(4) OVER (ORDER BY Last\_Order\_Date) AS RFM\_Recency,

ntile(4) OVER (ORDER BY Order\_Count) as RFM\_Frequency,

ntile(4) OVER (ORDER BY Average\_Order\_Value) as RFM\_Monetary

FROM (

SELECT c.customername AS Customer\_Name,

MAX(o.orderdate) AS Last\_Order\_Date,

COUNT(\*) AS Order\_Count,

SUM(ol.orderedquantity \* ol.extendedprice)/ COUNT(\*) AS Average\_Order\_Value

FROM furniture.orders o

NATURAL JOIN furniture.orderline ol

NATURAL JOIN furniture.customer c

GROUP BY c.customername

)

)

ORDER BY RFM\_Recency \* 100 + RFM\_Frequency \* 10 + RFM\_Monetary DESC;

**Results:**

|  |  |
| --- | --- |
| **CUSTOMER\_NAME** | **RFM\_COMBINED** |
| ABC Furniture Co. | 442 |
| Impressions | 423 |
| Dunkins Furniture | 414 |
| Ikards | 343 |
| New Furniture | 342 |
| Contemporary Casuals | 321 |
| A Carpet | 234 |
| Value Furnitures | 231 |
| Wild Bills | 224 |
| Janet's Collection | 222 |
| Home Furnishings | 131 |
| Flanigan Furniture | 113 |
| Furniture Gallery | 112 |
| Eastern Furniture | 111 |

**Justification:**

I used Recency Frequency Monetary (RFM) to evaluate who is the “best” customers using past purchase behavior. After joining 3 tables (ORDERS, ORDERLINE, CUSTOMER), I calculated 3 scores based on how recently the customer purchased (recency), how often they make purchases (frequency) and how much they spend in dollars on average on each purchase (monetary). Customers can be segmented to 4 equal groups by using percentiles to score the customers along each of the dimensions. After assigning a score between 1 and 4 for each, an aggregate score can be combined.

The indices put more emphasis on recency because repeat purchasers are more valuable and loyal than just one-time buyers, and those are the kind of customers the company needs to increase retention and lifetime value. According to the results, ABC Furniture Co. has the highest combined score, hence the most active and high value customer.

**8) Which Product Line has the largest percentage Month-Over-Month sales (total dollar amount sold)? (Hint: Look at the LAG function)**

SELECT Productline\_Name, Each\_Month\_of\_Each\_Year, Total\_Dollars, Prior\_Total\_Dollars,

(Total\_Dollars - Prior\_Total\_Dollars)/ Prior\_Total\_Dollars AS Percentage\_Month\_over\_Month\_Sales

FROM (

SELECT pl.productlinename AS Productline\_Name,

TO\_CHAR(o.orderdate, 'YYYY-MM') AS Each\_Month\_of\_Each\_Year,

SUM(ol.orderedquantity \* ol.extendedprice) AS Total\_Dollars,

LAG(SUM(ol.orderedquantity \* ol.extendedprice), 1, NULL) OVER

(ORDER BY pl.productlinename, TO\_CHAR(o.orderdate, ‘YYYY-MM’))

AS Prior\_Total\_Dollars

FROM furniture.orders o

NATURAL JOIN furniture.orderline ol

NATURAL JOIN furniture.product p

NATURAL JOIN furniture.productline pl

GROUP BY pl.productlinename,

TO\_CHAR(o.orderdate, ‘YYYY-MM’)

)

ORDER BY (Total\_Dollars – Prior\_Total\_Dollars)/ Prior\_Total\_Dollars DESC

FETCH FIRST 2 ROWS ONLY;

**Results:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PRODUCTLINE\_NAME** | **EACH\_MONTH\_OF\_EACH\_YEAR** | **TOTAL\_DOLLARS** | **PRIOR\_TOTAL\_DOLLARS** | **PERCENTAGE\_MONTH\_OVER\_MONTH\_SALES** |
| Antique | 2015-01 | 600 |  |  |
| Basic | 2016-12 | 14330 | 1055 | 12.58293838862559241706161137440758293839 |

**9) Create a VIEW in your own schema that joins all of the columns in these tables: PRODUCTLINE, PRODUCT, CUSTOMER, ORDERS, ORDERLINE, CUSTOMERSHIPADDRESS, TERRITORY and SALESPERSON. Be aware of Cartesian products e.g., between the ship address and customer.**

CREATE VIEW v1

AS

SELECT \*

FROM furniture.productline pl

NATURAL JOIN furniture.product p

NATURAL JOIN furniture.customer c

NATURAL JOIN furniture.orders o

NATURAL JOIN furniture.orderline ol

INNER JOIN furniture.customershipaddress cs USING(customerid)

INNER JOIN furniture.territory t USING(territoryid)

LEFT JOIN furniture.salesperson sp USING(salespersonid)

WHERE o.shipadrsid = cs.shipaddressid;

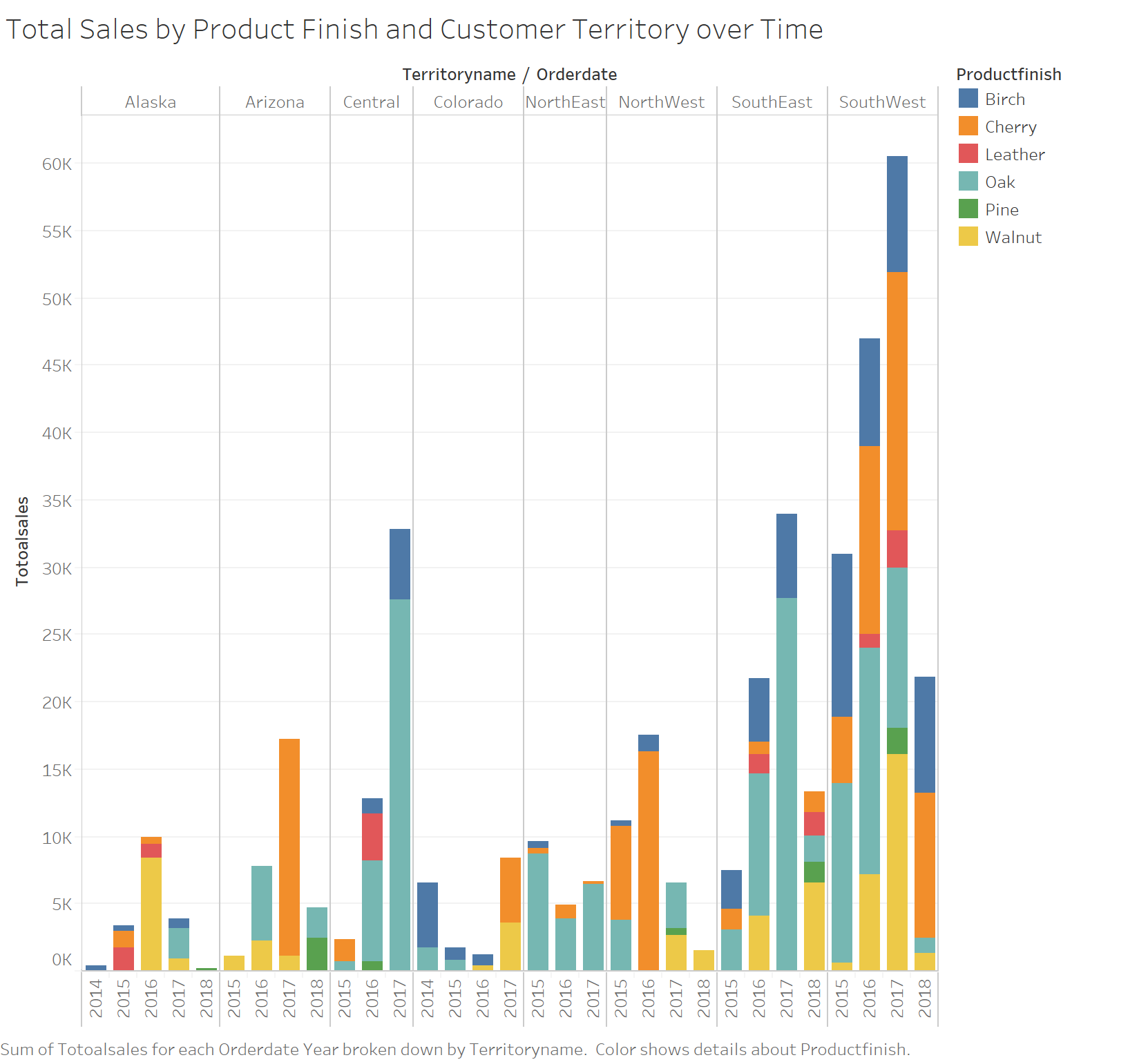
**10) Import all the data from your VIEW into Microsoft Excel. Create a pivot table from the resulting data set and then summarize the data according to total sales by product line and Customer State.**

**Results:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sum of TOTOALSALES** | **Product Line** | | | | |
| **Customer State** | **Antique** | **Basic** | **Classical** | **Modern** | **Grand Total** |
| CA | $1,800.00 | $22,065.00 |  | $4,936.00 | $28,801.00 |
| CO | $2,568.80 | $15,822.20 |  |  | $18,391.00 |
| FL | $19,692.00 | $7,492.00 | $6,000.00 | $5,597.00 | $38,781.00 |
| Il | $1,152.00 | $22,900.00 | $7,000.00 | $4,375.00 | $35,427.00 |
| NJ | $4,440.00 | $10,515.80 | $870.00 | $2,150.18 | $17,975.98 |
| NM | $13,620.00 | $57,044.40 | $880.00 | $11,419.80 | $82,964.20 |
| NY | $14,024.60 | $83,935.55 | $28,050.00 | $10,720.00 | $136,730.15 |
| TX | $3,600.00 | $22,831.00 |  | $3,715.00 | $30,146.00 |
| VA | $10,425.00 | $925.00 | $5,500.00 | $3,631.00 | $20,481.00 |
| **Grand Total** | **$71,322.40** | **$243,530.95** | **$48,300.00** | **$46,543.98** | **$409,697.33** |

**11) Import all the data from your VIEW into Tableau. Create an appropriate visualization from the resulting data set that summarizes the total sales by product finish and customer territory over time.**

**Results:**



**12) How many hours did you spend working on the assignment? What was the most difficult part of completing the assignment?**

I did not really keep track of the time because I did not finish it at one time. I would say I spend around 10 hours to do it. This is the first assignment and I had to admit I already forgot some of the SQL. So, I did spend some extra time looking for code and reviewing the class recordings first then started doing the assignment. I think the difficult part was that it was easy to get errors in Oracle SQL Developer and I could not really tell where I did wrong, so I had to spend some time check the code. Besides, the questions are not as easy as the introductory database course, but they are also challenging and applicable to practical situations.