Assignment 13 Name Oliver Conover

Using the AdventureWorks2012 database for all questions. (30 points)

1. Using the Sales.SalesOrderHeader table, (example page 453) (13 pts)

a. How many unique territoryids are there?

10

b. Paste query here

select distinct TerritoryID

from sales.SalesOrderHeader soh

c. how many salesorders are there?

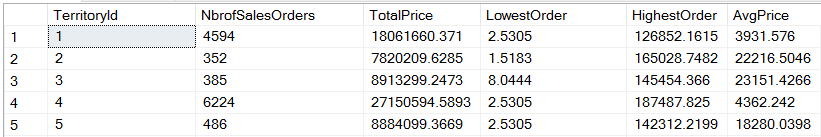
31465

d. Paste your query here.

select SalesOrderID

from sales.SalesOrderHeader soh

e. Write a query to produce the following results. In your result set, be sure to match the **column titles that are shown in the example below**. **Sort by territoryid**. For all calculations, use the TotalDue field. (6 pts) The example below only shows the first five rows of your result set.



Paste your query here.

select soh.TerritoryID, count(soh.SalesOrderID) NbrofSalesOrders, sum(soh.TotalDue) TotalPrice, min(soh.TotalDue) LowestOrder,

max(soh.TotalDue) HighestOrder, avg(soh.TotalDue) AvgPrice

from sales.SalesOrderHeader soh

group by soh.TerritoryID

order by soh.TerritoryID

f. How many rows did your query return? 10 Is that the same as your answer for # 1 a? yes

g. Looking at your result set, add up the NbrofSalesOrders column on a calculator, what is the total? Is that the same as your answer to # 1c? yes

31465

h. Link in the Sales.SalesTerritory table and change the query to return the Territory ID and Name. I still want the results sorted by territory id, so you have the same first four rows as shown in step 1.e. Hint, you will need to change the group by statement. (4 pts)

Paste your query here.

select soh.TerritoryID, st.Name, count(soh.SalesOrderID) NbrofSalesOrders, sum(soh.TotalDue) TotalPrice, min(soh.TotalDue) LowestOrder,

max(soh.TotalDue) HighestOrder, avg(soh.TotalDue) AvgPrice

from sales.SalesOrderHeader soh, sales.SalesTerritory st

where soh.TerritoryID = st.TerritoryID

group by soh.TerritoryID, st.Name

order by soh.TerritoryID

2. Using the Sales.SalesOrderHeader table, (example page 455) (4 pts)

a. Write a query to show the SalesPersonID and the last date they placed an order (using the OrderDate field) use an alias of LastOrderDate.

Paste query here.

select soh.SalesPersonID, max(soh.OrderDate) LastOrderDate

from Sales.SalesOrderHeader soh

group by soh.SalesPersonID

b. Change the query to show their first orderdate with an alias of FirstOrderDate. Paste your query here.

select soh.SalesPersonID, min(soh.OrderDate) FirstOrderDate

from Sales.SalesOrderHeader soh

group by soh.SalesPersonID

c. Eliminate the NULL salesperiodid from your last query. Paste your query here.

select soh.SalesPersonID, min(soh.OrderDate) LastOrderDate

from Sales.SalesOrderHeader soh

where soh.SalesPersonID is not null

group by soh.SalesPersonID

d. How many rows are in your result set?

17

3. Use the salesorderheader.salespersonid to link with the person.person table using the businessentityid. Only Paste your query for the last step. (6 pts)

a. Write a query to link the two tables listed about. Should get 3806 rows.

select \*

from Sales.SalesOrderHeader soh, Person.Person p

where soh.SalesPersonID = p.BusinessEntityID

b. For your Select statement include the sales persons first and last name in one field (not two fields) with a space between the first and last name using an Alias of FullName.

c. Add to your Select Statement, the lowest value of TotalDue using an alias of SmallestOrder. Sort the result set by this SmallestOrder. Paste your query here.

select p.FirstName + ' ' + p.LastName FullName, min(soh.TotalDue) SmallestOrder

from Sales.SalesOrderHeader soh, Person.Person p

where soh.SalesPersonID = p.BusinessEntityID

group by p.FirstName, p.LastName

order by SmallestOrder

4. Using the Sales.SalesOrderHeader table, show how many orders have a blank credit card id. (2 pts)

a. Paste your query here.

select soh.SalesOrderID

from Sales.SalesOrderHeader soh

where soh.CreditCardID is null

b. How many rows does your result set contain?

1131

c. Change your query to total the amount of freight paid on sales orders without credit card numbers. Paste your query here.

select sum(soh.Freight) TotalFreightCost

from Sales.SalesOrderHeader soh

where soh.CreditCardID is null

d. What is the freight amount?

$34,893.85

5. Using the Sales.SalesOrderHeader and Sales.SalesTerritory tables, by Territory name, what is the minimum, maximum and average amount of **tax** on all sales orders. Assign appropriate **alias names** to the three fields. Hint, you should have the same number of rows as questions 1 a. (5 pts)

a. Paste your query here.

select soh.TerritoryID, st.Name, min(soh.TaxAmt) LowestTax, max(soh.TaxAmt) HighestTax, avg(soh.TaxAmt) AverageTax

from Sales.SalesOrderHeader soh, Sales.SalesTerritory st

where soh.TerritoryID = st.TerritoryID

group by soh.TerritoryID, st.Name