Assignment 14 Name Oliver Conover

Use the AdventureWorks2012 database for all questions. (25 points)

1. Using only the Sales.SalesOrderDetail table, (10 pts)

a. Using the **distinct** function in your query, how many unique productid are in your result set?

266

b. Paste query here

select distinct sod.ProductID

from Sales.SalesOrderDetail sod

c. Remove the word “distinct” from your query and add a GROUP by for productid. How many rows are in your result set?

266

d. Paste your query here.

select sod.ProductID

from Sales.SalesOrderDetail sod

group by sod.ProductID

e. Changing the query from the previous step. Count how many times each productid is in the SaleOrderDetail table. Show the productid and the count. Paste your query here.

select sod.ProductID, count(sod.ProductID) ProductInv

from Sales.SalesOrderDetail sod

group by sod.ProductID

f. How many rows did your query return? Is that the same as your answer for # 1 a? Note: Using a distinct or Group by, both elimate duplicate productids.

266

g. Give your count an alias. Paste your query here.

select sod.ProductID, count(sod.ProductID) ProductInv

from Sales.SalesOrderDetail sod

group by sod.ProductID

h. Adding a HAVING Statement, show the products that are on less than 100 orders. Should have 38 rows. Paste your query here.

select sod.ProductID, count(sod.ProductID) ProductInv

from Sales.SalesOrderDetail sod

group by sod.ProductID

having count(sod.ProductID) < 100

order by ProductInv desc

i. Sort the result set by your count descending. Paste your query here.

see above

2. Using the SalesOrderHeader, show the number of orders for each sales person who has more than 100 orders. Do not show the salespersonids that are NULL value in your result set. (4 points)

a. Paste your query here.

select soh.SalesPersonID, count(soh.SalesOrderID) SalesPerPerson

from Sales.SalesOrderHeader soh

where soh.SalesPersonID is not null

group by soh.SalesPersonID

having count(soh.SalesOrderID) > 100

b. How many rows are in your result set?

13

c. To show that your number of rows, from the previous step is reasonable, write a simple query to show the unique SalesPersonID in the SalesOrderHeader. Paste your query here

select distinct soh.SalesPersonID

from Sales.SalesOrderHeader soh

d. How many rows are in your result set from step C?

18

Note: Notice the previous step included NULL. There are four salespersons with less than 100 orders, so the difference between step b and step c should be 4 (if you ignore the NULL row).

3. Using the Production.ProductCostHistory table, show the products that have changed prices. Hint: If the product is only listed once in this table, then it is still at its original price. (7 points)

a. Paste your query here.

select pch.ProductID, count(pch.ProductID) PriceChanges

from Production.ProductCostHistory pch

group by pch.ProductID

having count(pch.ProductID) > 1

b. Change your query to list the products that have changed cost more times than 2 times. Paste your query here. Should have 25 rows in your result set.

select pch.ProductID, count(pch.ProductID) PriceChanges

from Production.ProductCostHistory pch

group by pch.ProductID

having count(pch.ProductID) > 2

c. Revising the query from the previous step. Link in the Production.Product table, so you can include the product name in your result set. You should still have 25 rows in your result set. Paste your query here.

select pch.ProductID, pro.Name, count(pch.ProductID) PriceChanges

from Production.ProductCostHistory pch, Production.Product pro

where pro.ProductID = pch.ProductID

group by pch.ProductID, pro.Name

having count(pch.ProductID) > 2

4. From your query in the previous step, output your result set to an excel file: (4 pts)

* Make sure each column has a title (either by using an alias or by typing in excel)l
* File name is your last name and A4.

Make sure that I can just click on your file and it opens in excel looking good. Include this file with your assignment upload.