[**PROG8590-24F-Sec1-Relational**](https://conestoga.desire2learn.com/d2l/home/1255613) **Databases**

**NAME: VETRI VELAN ANANDAN**

**STUDENT ID: 8979936**

**Assignment No: 3**

**Date : 8/10/2024**

1. *We need to know the number of products we have in the PurchaseOrderDetail table. (****count the number*** *of un-repeated productid)*

*Answer:*

**SELECT COUNT(DISTINCT [ProductID])**

*//We are selecting the count , with distinct or non-repeated product Ids.*

**FROM [Purchasing].[PurchaseOrderDetail]**

*// choosing the table*

*Output:*

A screenshot of a computer

Description automatically generated

*2. Write a query to show the productID of the most profitable product(ignore production costs) after price and order quantity are considered (maximum amount of money gained for each product id)*

*•* ***Use SUM and group by*** *to get the best result.*

*•* ***HINT: Should be 3358797.75***

*Answer:*

**SELECT TOP 1 [ProductID]**

*// Selecting only 1 , that comes the top of the list of product ids.*

**FROM (**

**SELECT [ProductID], SUM( [UnitPrice]\* [OrderQty]) AS total\_profit**

*// mentioning the SUM, unit price\*order quantity as total profit*

**FROM [Purchasing].[PurchaseOrderDetail]**

*//Choosing the table*

**GROUP BY [ProductID]**

**) AS product\_profits**

*// Grouping as product profit*

**ORDER BY total\_profit DESC;**

// ordering the output in descending order.

*Output:*

A computer screen shot of a computer

Description automatically generated

*3. Write a query to show the names of the top 5 most profitable products, as in question 2. Remember to take both price and quantity sold into account.*

*•* ***You must join two tables****.*

*Answer:*

**SELECT TOP 5 [Name] , SUM( [OrderQty] \* [UnitPrice] ) AS profit**

*//choosing top 5 on the list to display and creating the SUM formula*

**FROM [Production].[Product] p**

*//Choosing the table*

**JOIN [Purchasing].[PurchaseOrderDetail] pod ON p.[ProductID] = pod.[ProductID]**

*//adding joins between the tables with common key to create relationships*

**GROUP BY p.[Name]**

**ORDER BY profit DESC;**

*//Grouping the names and ordering the profit in descending order*

*Output:*

A screenshot of a computer

Description automatically generated

*4. Write a query to show all product ID with the stock quantity less than average stock quantity.*

*• You have to use sub query*

*Answer:*

**SELECT [ProductID],**

//selecting the product id to be displayed

**(SELECT TOP 1 [StockedQty]**

//*choosing top 1 as the return will be more than 1*

**FROM [Purchasing].[PurchaseOrderDetail]**

*//choosing the table*

**WHERE [Purchasing].[PurchaseOrderDetail].ProductID =[Production].[Product].ProductID) AS stockQuantity**

*//creating the subquery and mentioning the matching keys*

**FROM [Production].[Product]**

*//Choosing the table*

**WHERE ProductID IN (**

**SELECT ProductID**

**//** *choosing the other table and column*

**FROM [Purchasing].[PurchaseOrderDetail]**

**WHERE[StockedQty] < (SELECT AVG([StockedQty])**

*//Mentioning the operation where the stocked Qty is < than the average Stocked Qty by using the AVG built-in function or function*

**FROM[Purchasing].[PurchaseOrderDetail] )**

);

*Output:*

A screenshot of a computer

Description automatically generated

*5. We need to know the product id and the modified date of the products with special offer* ***“Half-Price Pedal Sale”.***

*Answer:*

**SELECT [ProductID], [ModifiedDate]**

*// Selecting the columns to be displayed*

**FROM [Sales].[SpecialOfferProduct]**

*//choosing the table*

**WHERE [SpecialOfferID] = (**

**SELECT [SpecialOfferID] FROM [Sales].[SpecialOffer]**

**//***Mentioning the restriction on the special offer detials form the table*

**WHERE [Description] = 'Half-Price Pedal Sale');**

*//input to bring out the required value, that is to get only items under the offer ‘ half price pedal sale.*

*Output:*

A screenshot of a computer

Description automatically generated