***Assignment NO. - 3***

***Student Name: Basavraj Nagappa Jaliminche***

***Student Id: 8800149***

***PROGRAM: Reporting Systems and Database Development (1517)***

***Course Name: Relational Databases PROG 8590***

***Assignment NO. - 3***

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

Query:

USE AdventureWorks2014

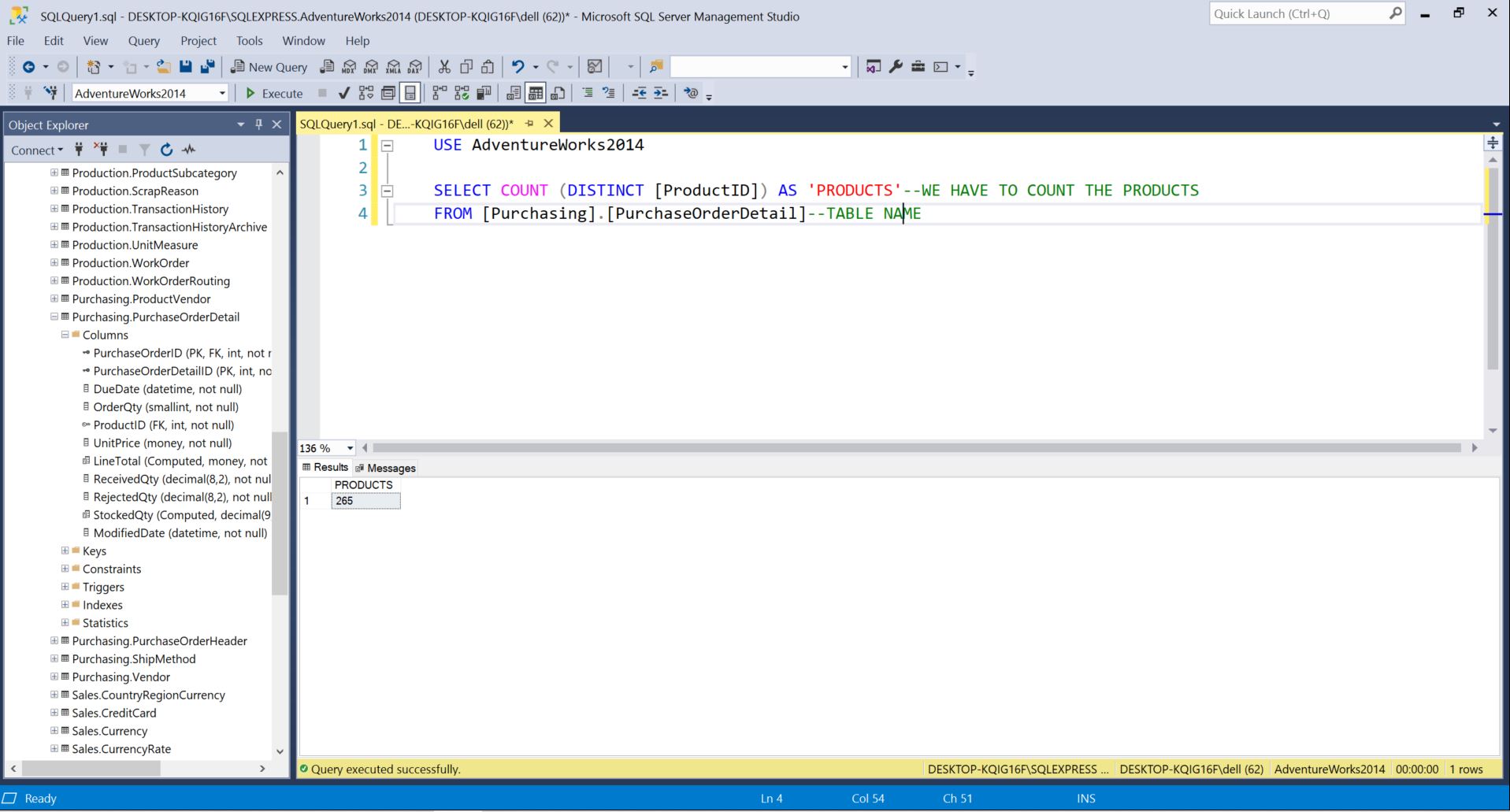
SELECT COUNT (DISTINCT [ProductID]) AS 'PRODUCTS'--WE HAVE TO COUNT THE PRODUCTS

FROM [Purchasing].[PurchaseOrderDetail]--TABLE NAME

Description:

* We have to select the products from the table [Purchasing].[PurchaseOrderDetail]
* We have to count the number of products so we have used the count function and the products does not duplicate so that we have to use DISTICT keyword.

Screenshot of the query is provided on the further page ..



**Query no 1**

Q.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

Query:

SELECT TOP 1 SUM([LineTotal]) AS 'TOTAL AMMOUNT',[ProductID]---SELECTED TOP 1LINE TOATL AND PRODUCT ID

FROM [Purchasing].[PurchaseOrderDetail]--FROM TABLE

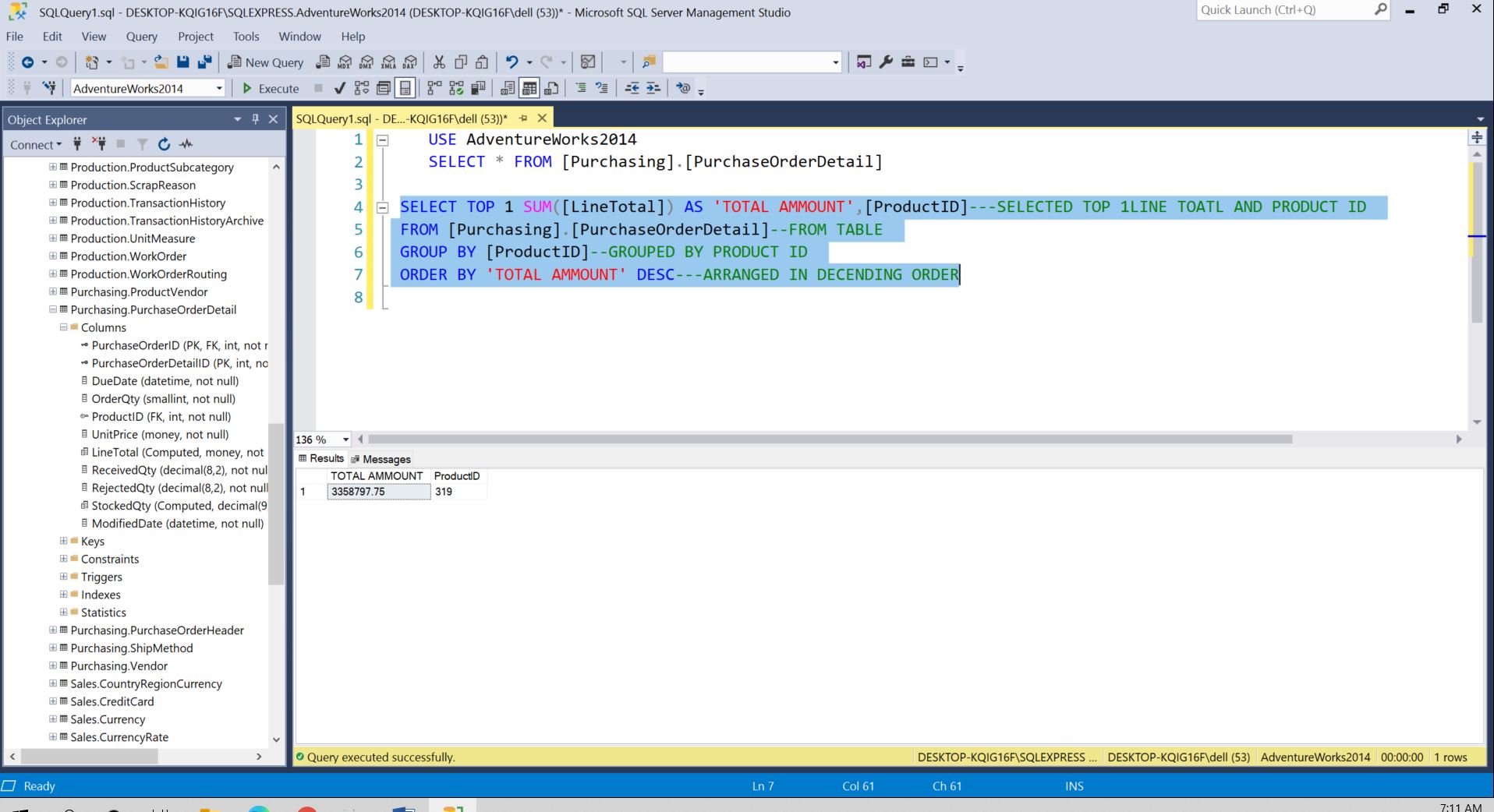
GROUP BY [ProductID]--GROUPED BY PRODUCT ID

ORDER BY 'TOTAL AMMOUNT' DESC---ARRANGED IN DECENDING ORDER

Description:

* We have selected table [Purchasing].[PurchaseOrderDetail]
* We have made sum of line total and then made group and we have arranged the result in the descending manner.
* After that we have selected the top most of the result along with the product id.

Screenshot of the query is provided on the further page ..



**Query 2**

Q.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.

Query:

SELECT TOP 5 PO.ProductID,P.Name AS 'PRODUCT NAME',SUM(PO.[LineTotal]) AS 'TOTAL AMMOUNT'--SELECTED

--TOP 5 PRODUCTS WITH THE MAXIMUM PROFITS

FROM [Purchasing].[PurchaseOrderDetail] PO

INNER JOIN [Production].[Product] P --JOINING THE TWO TABLES WITH INNER JOIN

ON PO.ProductID = P.ProductID --LINK BETWEEN THE TABLES

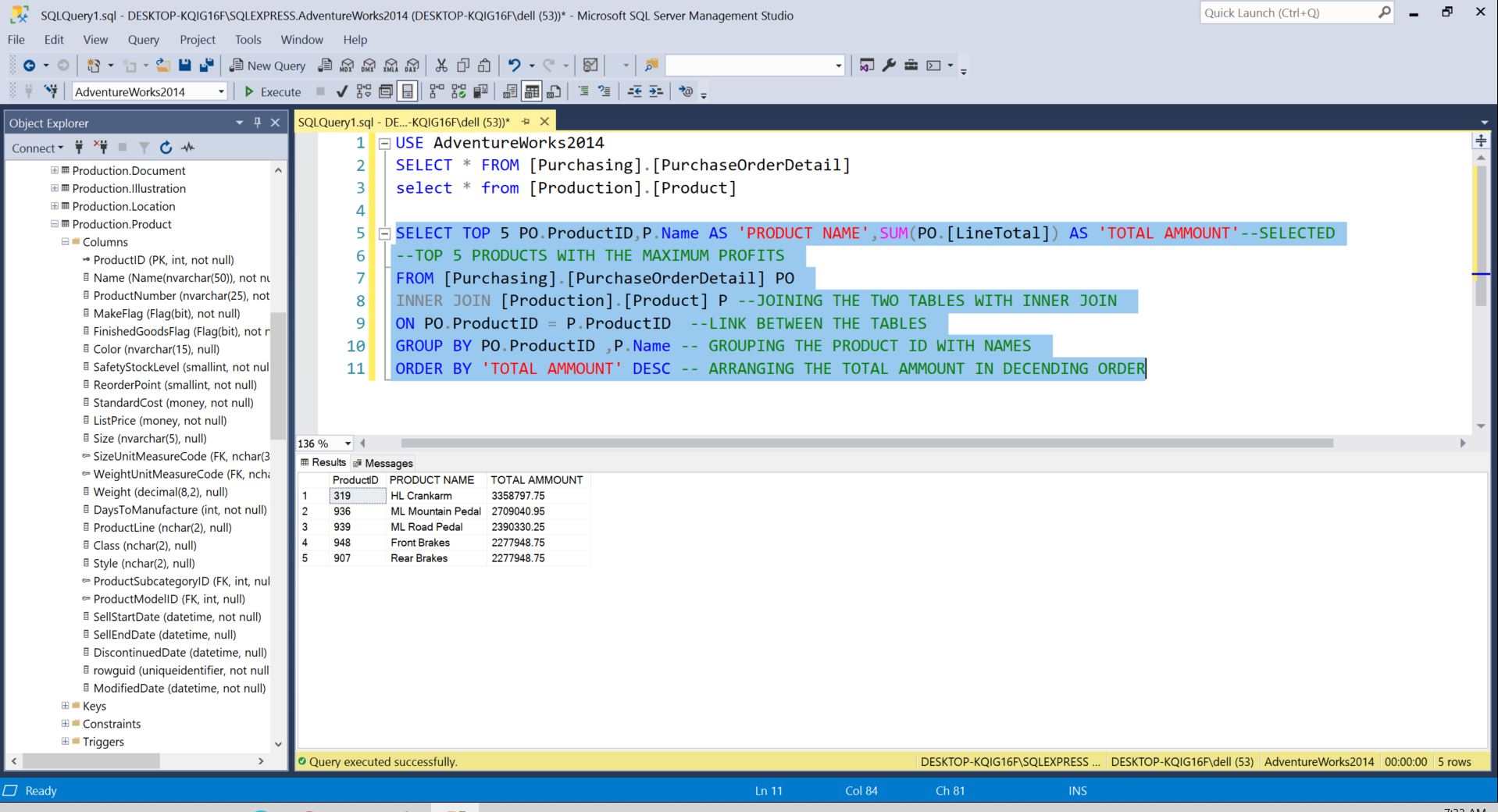
GROUP BY PO.ProductID ,P.Name -- GROUPING THE PRODUCT ID WITH NAMES

ORDER BY 'TOTAL AMMOUNT' DESC -- ARRANGING THE TOTAL AMMOUNT IN DECENDING ORDER

Description:

* We have selected the two tables [Purchasing].[PurchaseOrderDetail] and Production].[Product]
* Then we have joined the two tables with the link ProductID.
* After joining the two tables we have groped the results with tha product id and product name.
* After that we have arranged the sum of line total in the descending order.
* For showing the results we hae selected the top 5 products with the product name and total amount.

Screenshot of the query is provided on the further page ..



**Query 3**

Q.4 Write a query to show all product ID with the stock quantity less than average stock quantity. • You have to use sub query.

Query:

SELECT [ProductID],[StockedQty]--DISPLAYING THE PRODUCT ID WITH STOCKED QTY

FROM [Purchasing].[PurchaseOrderDetail] --FROM TABLE

WHERE StockedQty > --CONDITION

(SELECT AVG(StockedQty)--- USING THE SUBQUERY

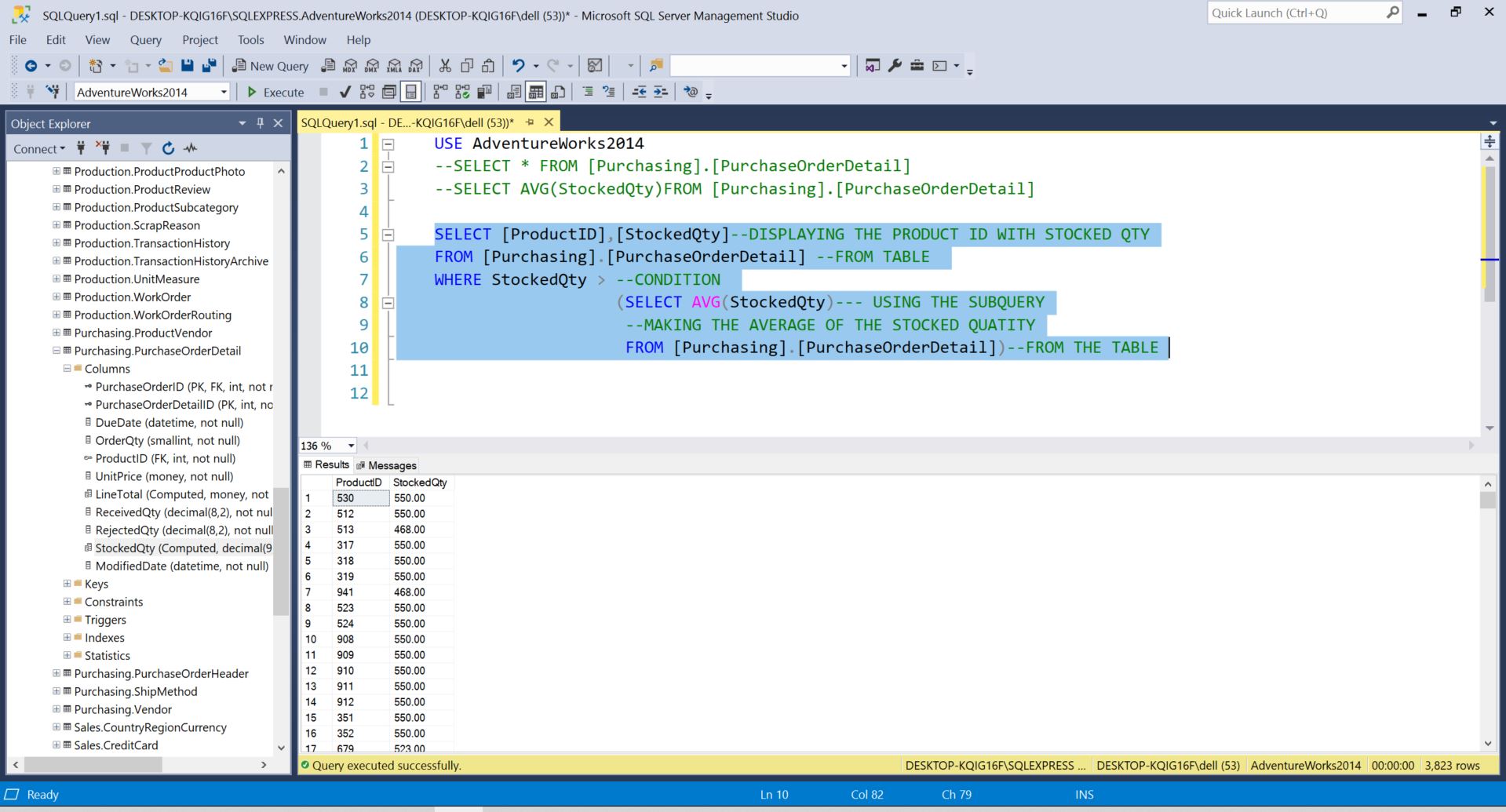
--MAKING THE AVERAGE OF THE STOCKED QUATITY

FROM [Purchasing].[PurchaseOrderDetail])--FROM THE TABLE

Description:

* We have select the table [Purchasing].[PurchaseOrderDetail].
* Then calculated average of the stocked qty.
* Then using subquery, we have given condition greater than the average stocked qty.
* Then we have displayed the results of product id with the stocked qty.

Screenshot of the query is provided on the further page ..



**Query 4**

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

Query:

SELECT SOP. ProductID,SOP.[ModifiedDate]--DISPLAYING PRODUCT ID WITH MODIFIED DATE

FROM [Sales].[SpecialOfferProduct] SOP

INNER JOIN [Sales].[SpecialOffer] SO --JOINING TWO TABLES WITH INNER JOIN

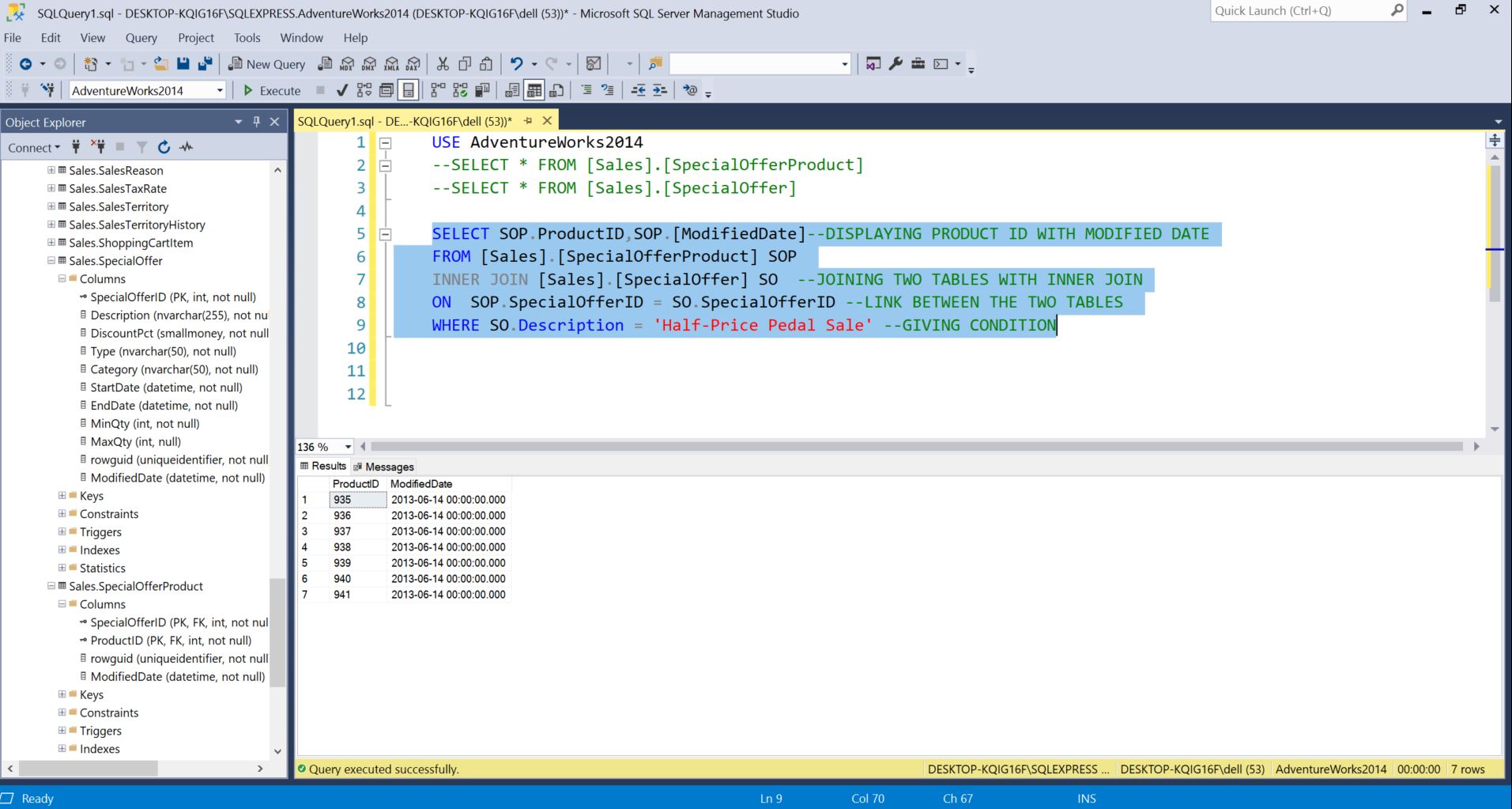
ON SOP.SpecialOfferID = SO.SpecialOfferID --LINK BETWEEN THE TWO TABLES

WHERE SO.Description = 'Half-Price Pedal Sale' --GIVING CONDITION

Description:

* We have selected two tables [SpecialOfferProduct] and [Sales].[SpecialOffer].
* Then we have joined them with inner join .
* Link between the two tables is SpecialOfferID.
* Then we have put condition as Description = 'Half-Price Pedal Sale'
* Then displayed product id with modified date.

Screenshot of the query is provided on the further page ..



**Query 5**