**Assignment 1 : Simple SQL Query with a single table with where clause.**

Create Database northwind\_database

Create table Products(

ProductID int NOT NULL PRIMARY KEY ,

ProductName VARCHAR(50) NOT NULL,

SupplierID int NOT NULL,

CategoryID int NOT NULL,

QuantityPerUnit VARCHAR(20) NOT NULL,

UnitPrice DECIMAL(10,4) NOt NUll,

UnitInStock SMALLINT NOT NULL,

UnitOnOrder SMALLINT NOT NULL,

ReorderLevel SMALLINT NOT NULL,

Discontinued BIT NOT NULL, )

Select \* from Products

INSERT INTO Products VALUES( 1 , 'chai' , '1' , '1' , '110 boxes x 20 bags' , '18.000' , '39' , '0' , '10' , '0')

INSERT INTO Products VALUES( 2 , 'chang' , '1' , '1' , '24 - 12 oz bottles' , '19.000' , '17' , '40' , '25' , '0')

INSERT INTO Products VALUES( 3 , 'Aniseed Syrup' , '1' , '2' , '12 - 550 ml bottles' , '10.000' , '13' , '70' , '25' , '0')

INSERT INTO Products VALUES( 4 , 'Chef Anton`s Cajun Seasoning' , '2' , '2' , '148 - 6 oz jars' , '22.000' , '53' , '0' , '10' , '0')

INSERT INTO Products VALUES( 5 , 'Chef Anton`s Gumbo Mix ' , '2' , '2' , '36 boxes' , '21.0350' , '0' , '0' , '10' , '1')

INSERT INTO Products VALUES( 6 , 'Grandma`s Boysenberry Spread' , '3' , '2' , '12 - 8 oz jars' , '25.000' , '120' , '0' , '25' , '0')

INSERT INTO Products VALUES( 7 , 'Uncle Bob`s Organic Dried Pears' , '3' , '7' , '12 - 1 lb pkgs.' , '30.000' , '15' , '0' , '10' , '0')

INSERT INTO Products VALUES( 8 , 'Northwoods Cranberry Sauce ' , '3' , '2' , '12 - 12 oz jars' , '40.000' , '6' , '0' , '0' , '0')

INSERT INTO Products VALUES( 9 , 'Mishi Kobe Niku' , '4' , '6' , '18 - 500 g pkgs' , '97.000' , '29' , '0' , '0' , '1')

INSERT INTO Products VALUES( 10 , 'Ikura ' , '4' , '8' , '12 - 200 ml jars' , '31.000' , '31' , '0' , '0' , '0')

INSERT INTO Products VALUES( 11 , 'Queso Cabrales' , '5' , '4' , '1 kg pkg' , '21.000' , '22' , '30' , '30' , '0')

INSERT INTO Products VALUES( 12 , 'Queso Manchego La Pastora' , '5' , '4' , '10 - 500 g pkgs' , '38.000' , '86' , '0' , '0' , '0')

INSERT INTO Products VALUES( 13 , 'Konbu' , '6' , '8' , '2 kg box' , '6.000' , '24' , '0' , '5' , '0')

INSERT INTO Products VALUES( 14 , 'Tofu' , '6' , '7' , '40 - 100 g pkgs' , '23.500' , '35' , '0' , '0' , '0')

INSERT INTO Products VALUES( 15 , 'Genen Shouyu' , '6' , '2' , '124 - 250 ml bottles' , '15.500' , '39' , '0' , '5' , '0')

SELECT ProductID , ProductName , UnitPrice

From Products where(Unitprice < 20)

SELECT ProductID , ProductName , UnitPrice

From Products where((Unitprice > 15) and (UnitPrice<25))

select distinct ProductName, UnitPrice from Products

where UnitPrice > (SELECT avg(UnitPrice) FROM Products)

select top 10 ProductName, UnitPrice from Products order by UnitPrice desc

select count(productName) from Products group by Discontinued

select top 10 ProductName, UnitPrice from Products order by UnitPrice desc

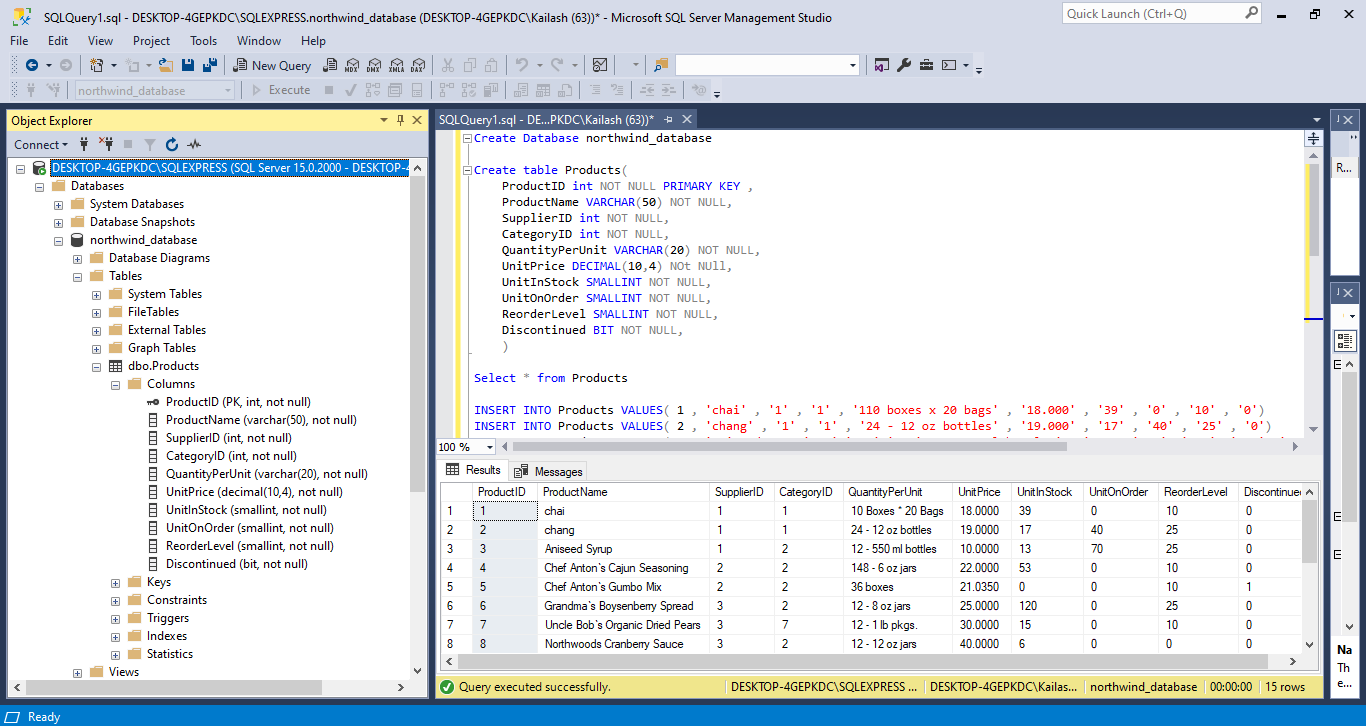


Figure : Object explorer window

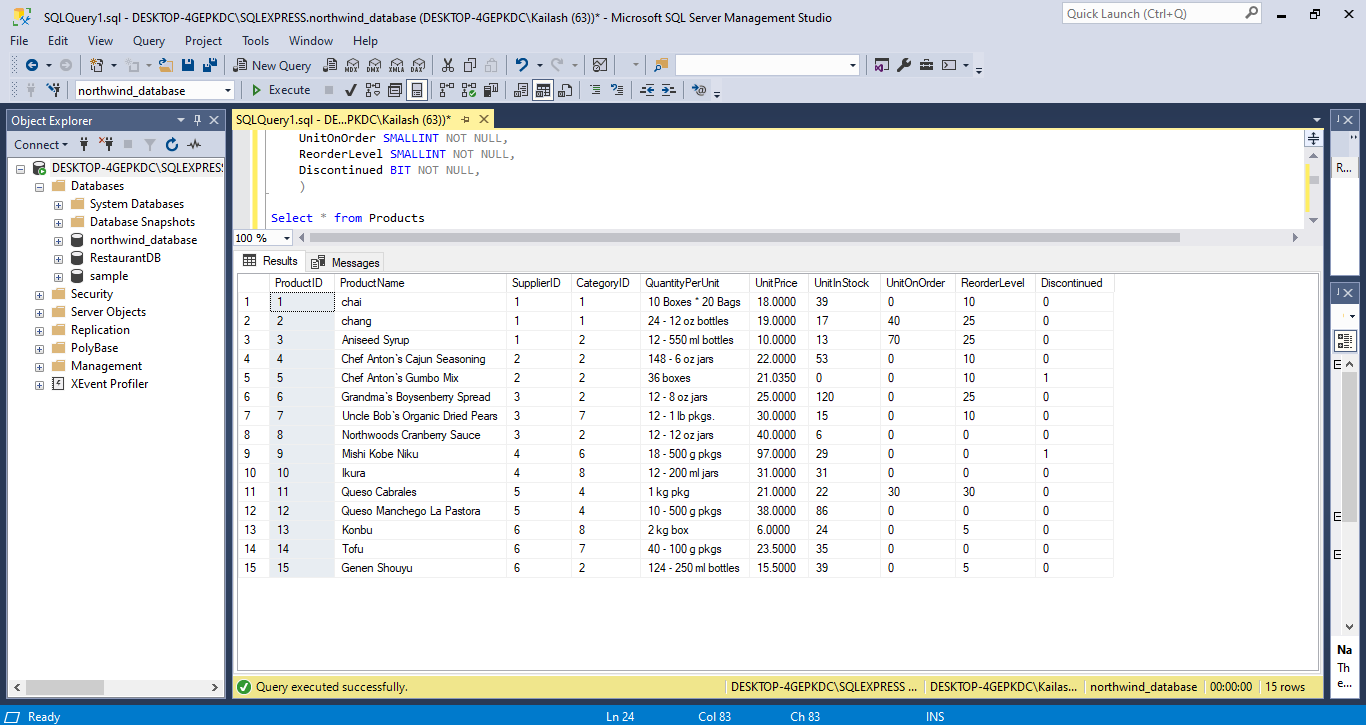


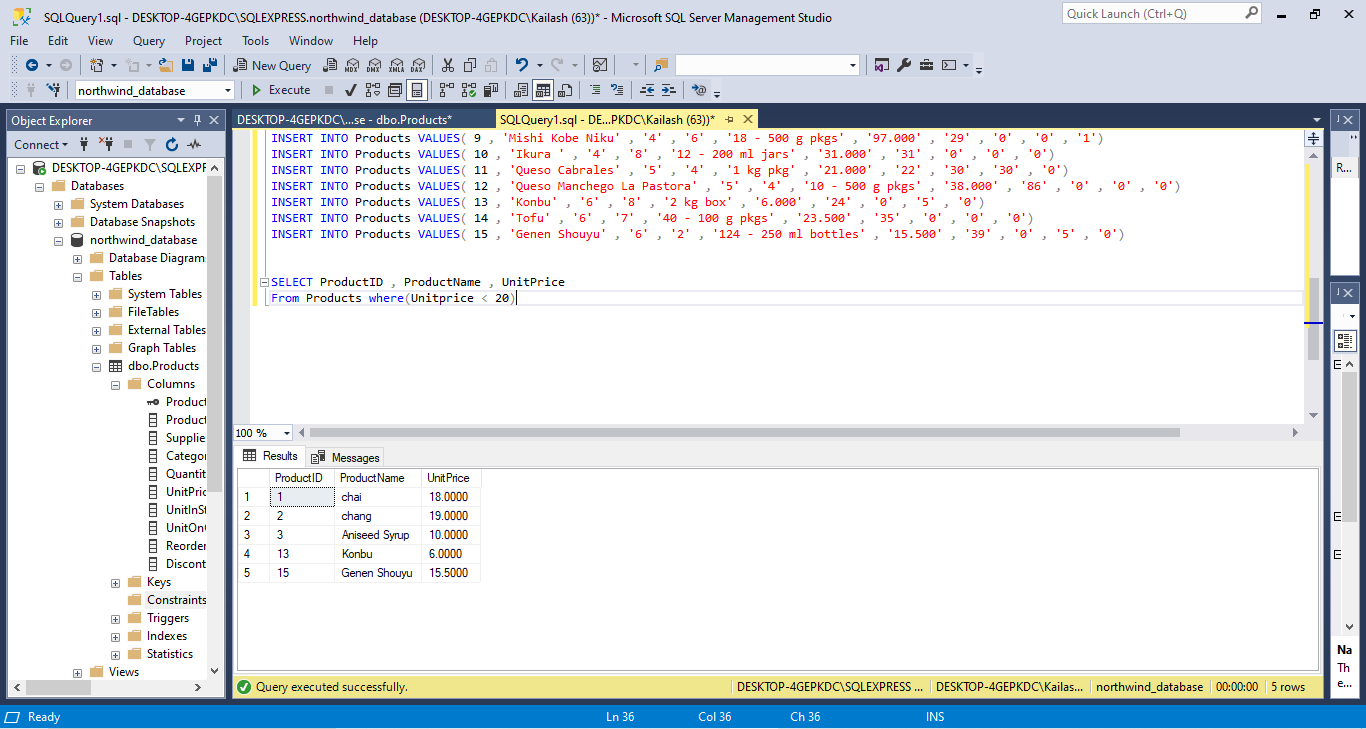
Figure: Output Table of Products

1. Write a query to get a Product list (id, name, unit price) where current products cost

less than $20.

SELECT ProductID , ProductName , UnitPrice

From Products where(Unitprice < 20)

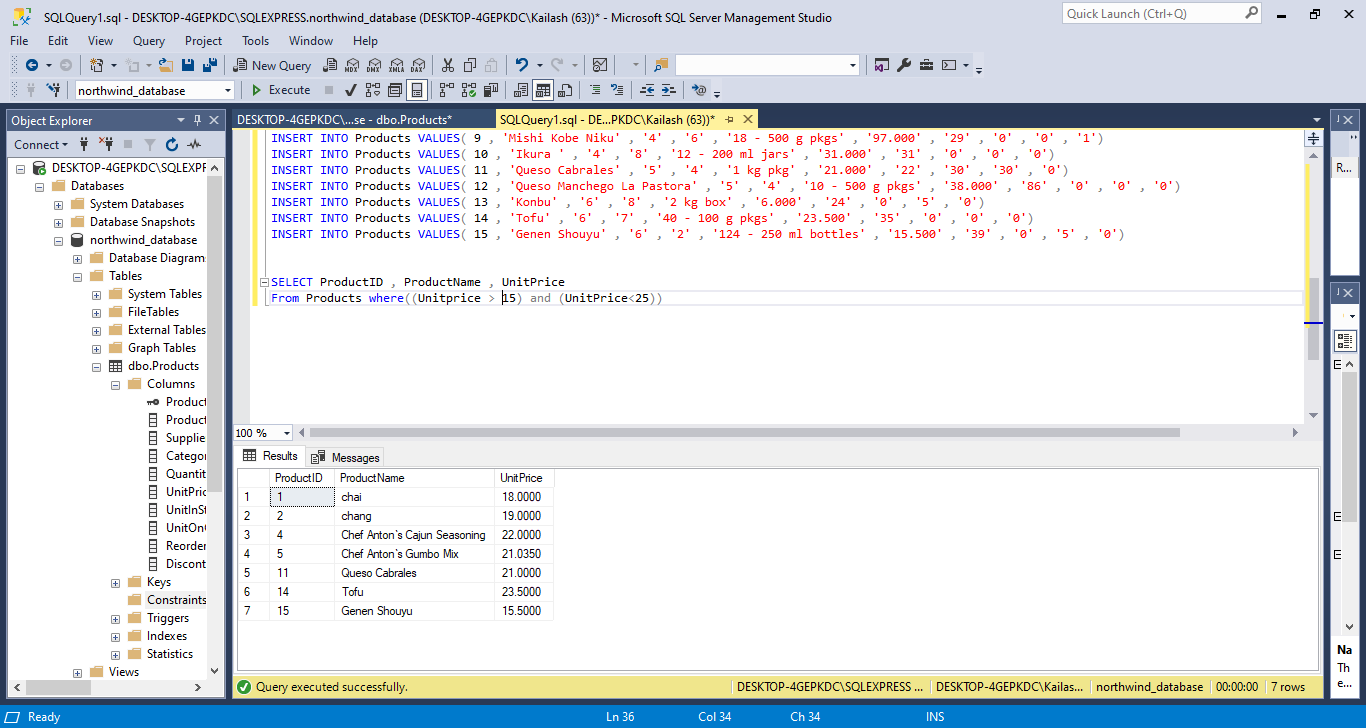


2. Write a query to get Product list (id, name, unit price) where products cost between

$15 and $25

SELECT ProductID , ProductName , UnitPrice

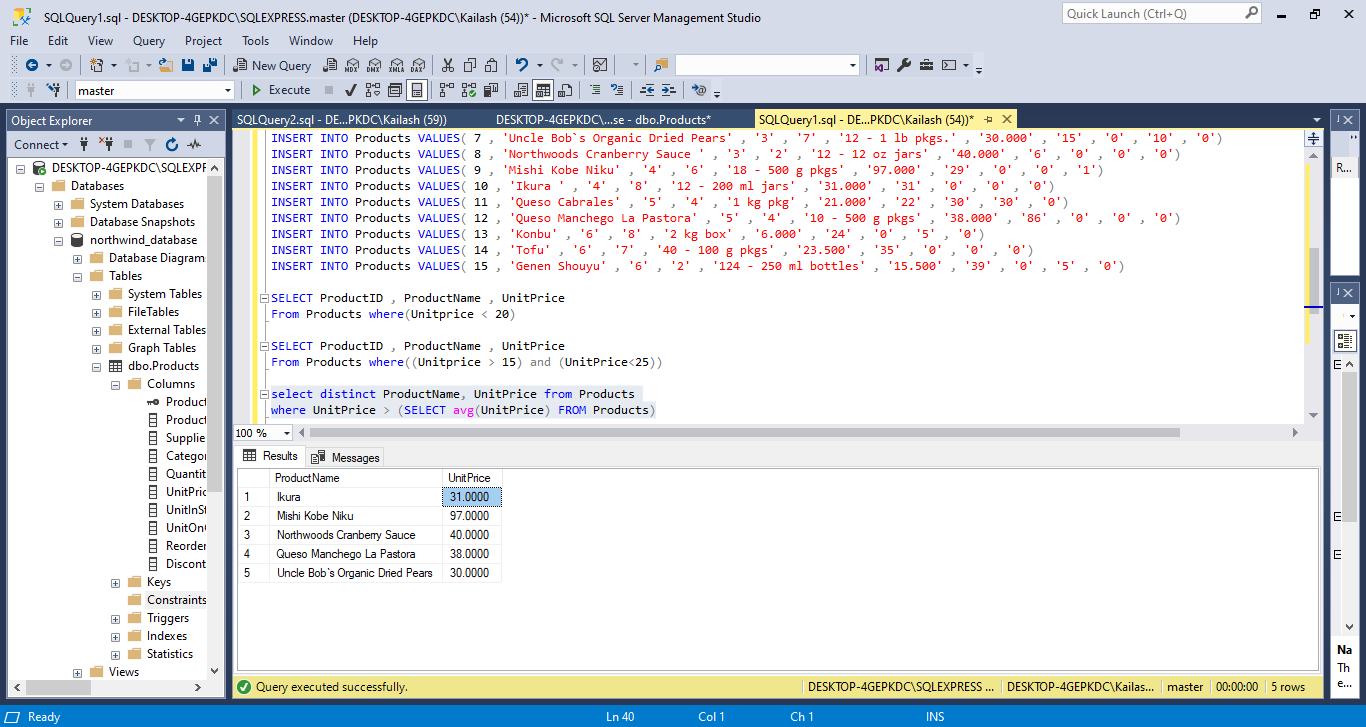
From Products where((Unitprice > 15) and (UnitPrice<25))



3. Write a query to get Product list (name, unit price) of above average price.

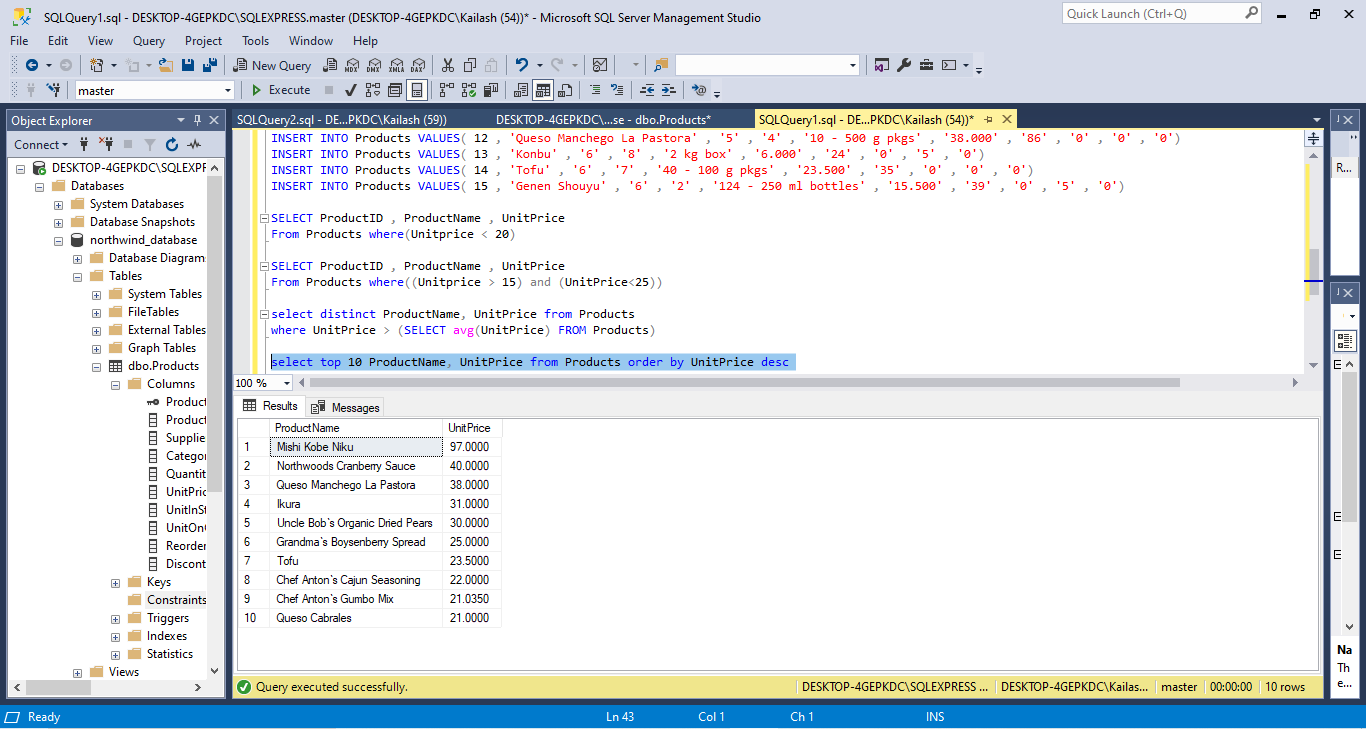
select distinct ProductName, UnitPrice from Products

where UnitPrice > (SELECT avg(UnitPrice) FROM Products)



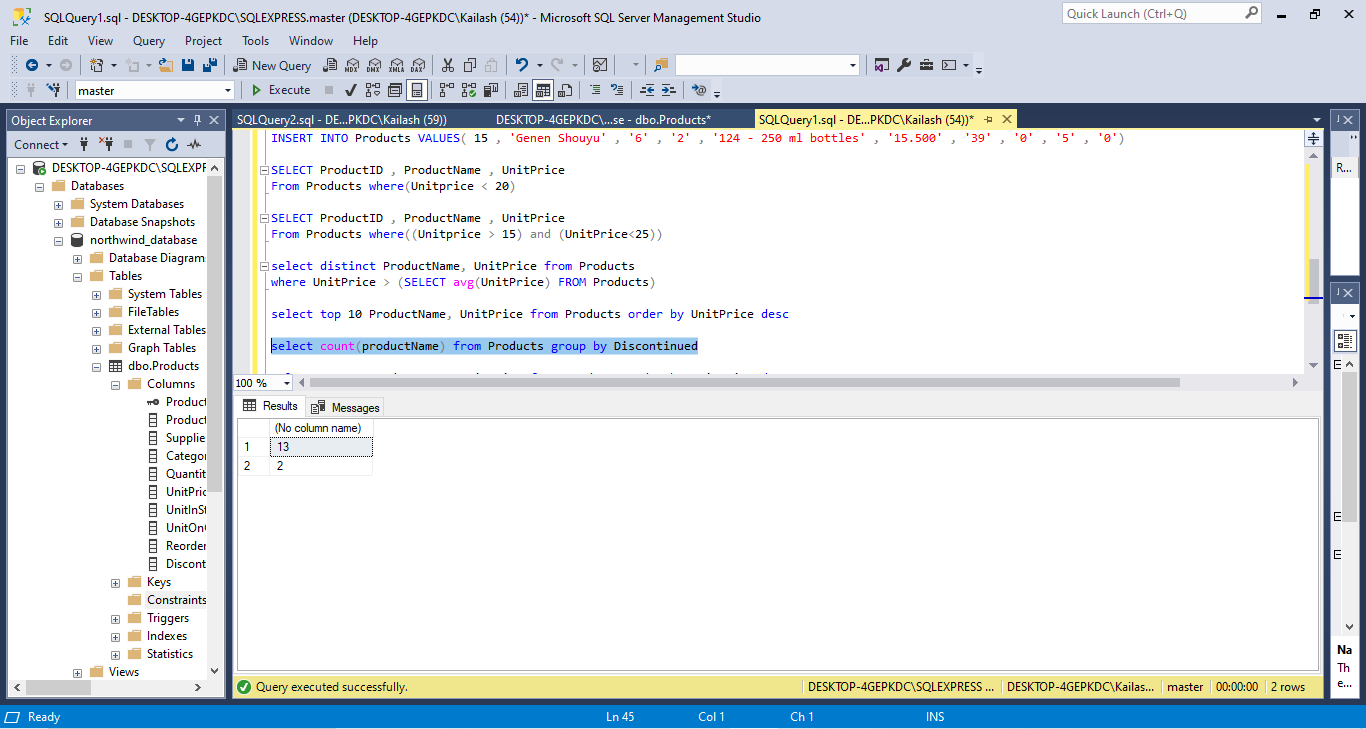
4. Write a query to get Product list (name, unit price) of ten most expensive products

select top 10 ProductName, UnitPrice from Products order by UnitPrice desc



5. Write a query to count current and discontinued products

select count(productName) from Products group by Discontinued



6. Write a query to get Product list (name, units on order , units in stock) of stock is less

than the quantity on order

select top 10 ProductName, UnitPrice from Products order by UnitPrice desc

