1)Create view OrderItemProducts that returns columns from the orders,orderitems,products tables

create view OrderItemProducts

as

select o.OrderID, o.OrderDate,o.TaxAmount,o.ShipDate,oi.ItemPrice,oi.DiscountAmount,oi.Quantity,

(oi.ItemPrice - oi.DiscountAmount) as FinalPrice,

(oi.Quantity \* (oi.ItemPrice - oi.DiscountAmount)) as ItemTotal,

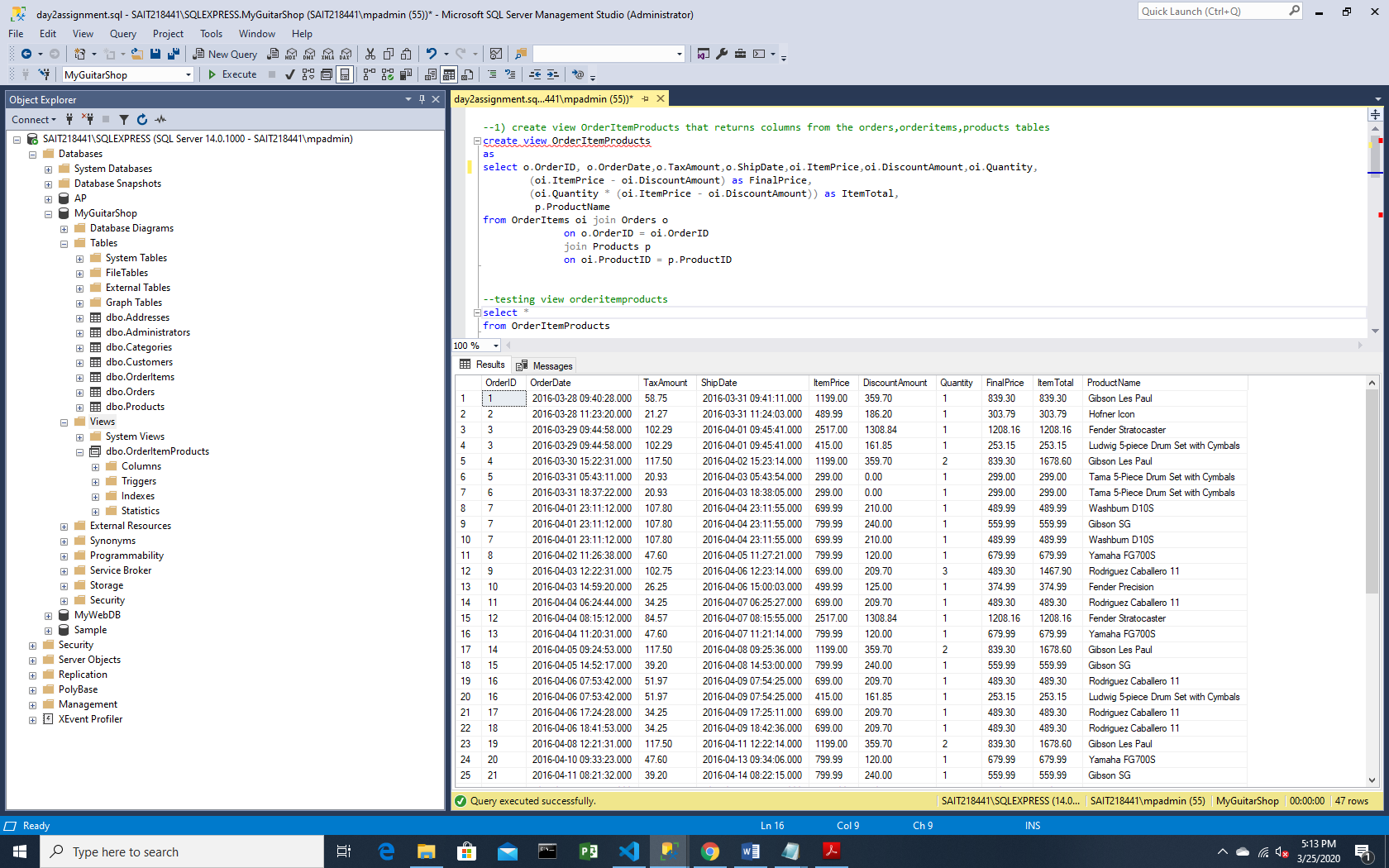
p.ProductName

from OrderItems oi join Orders o

on o.OrderID = oi.OrderID

join Products p

on oi.ProductID = p.ProductID



2)create view Prodcutsummary for the view orderitemproducts which returns productname,ordercount,ordertotal also display total sales (i.e , ordertotal) for the top 5 best selling products

create view ProductSummary

as

select ProductName,count(orderid) as ordercount, sum(quantity) as ordertotal

from OrderItemProducts

group by ProductName

select \*

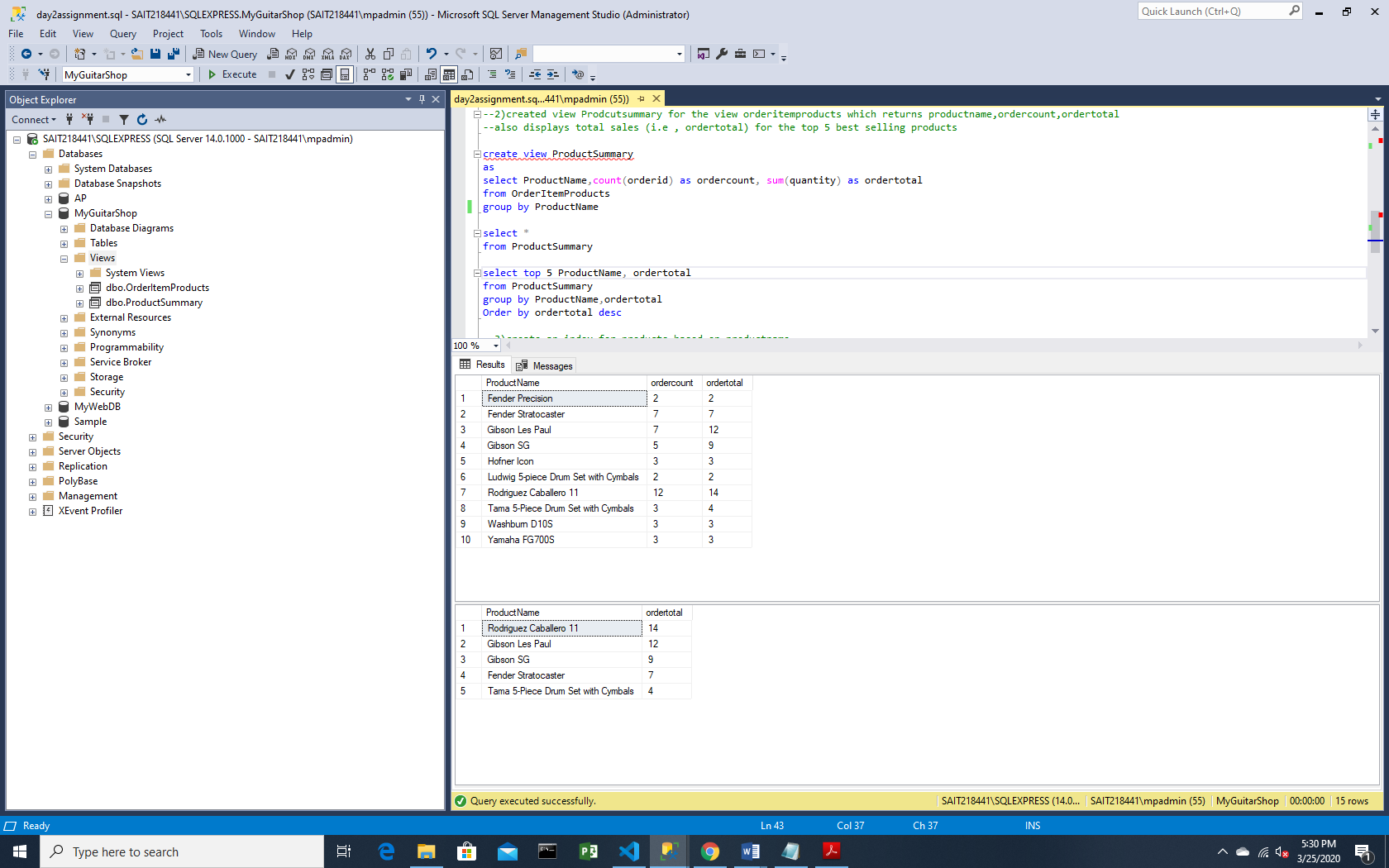
from ProductSummary

select top 5 ProductName, ordertotal

from ProductSummary

group by ProductName,ordertotal

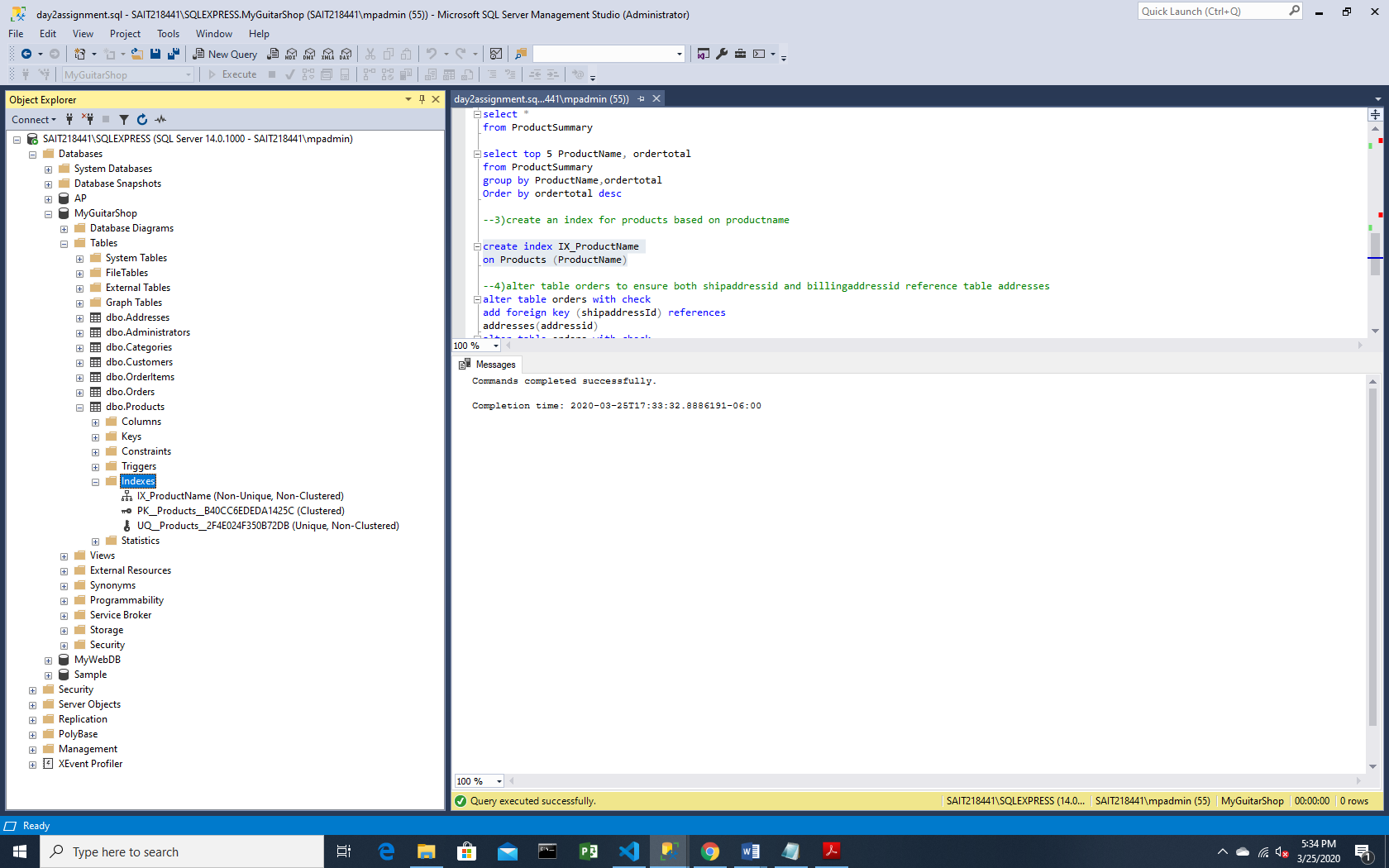
Order by ordertotal desc



3)create an index for products based on productname

create index IX\_ProductName

on Products (ProductName)



4)alter table orders to ensure both shipaddressid and billingaddressid reference table addresses

alter table Orders

add constraint Ord\_ShipAdrs\_FK foreign key (shipaddressId) references

Addresses(AddressID)

alter table Orders

add constraint Ord\_BillAdrs\_FK foreign key (billingaddressid) references

Addresses(AddressID)

