KULANICI TANIMLI FOKSİYONLAR (Devam)

Skaler değerli fonksiyonlara örnek:

Parametre müşteri grubu olacak, eğer grubu parametresi değeri NULL olursa tüm müşteri sayısını, parametre değeri NULL değilse verilen gruptaki müşteri sayısını veren bir fonksiyon oluşturunuz.



@grubu değeri NULL ise gerekli alt sorgu;

Select count(\*) from musteri

@grubu parametresi NULL değil ise gerekli alt sorgu

Select count(\*) from musteri where grubu=@grubu

CREATE FUNCTION fn\_musteri\_sayi (@grubu varchar(10))

RETURNS int

AS

Begin

Declare @sayi int

IF @grubu is null

Select @sayi = (Select count(\*) from musteri)

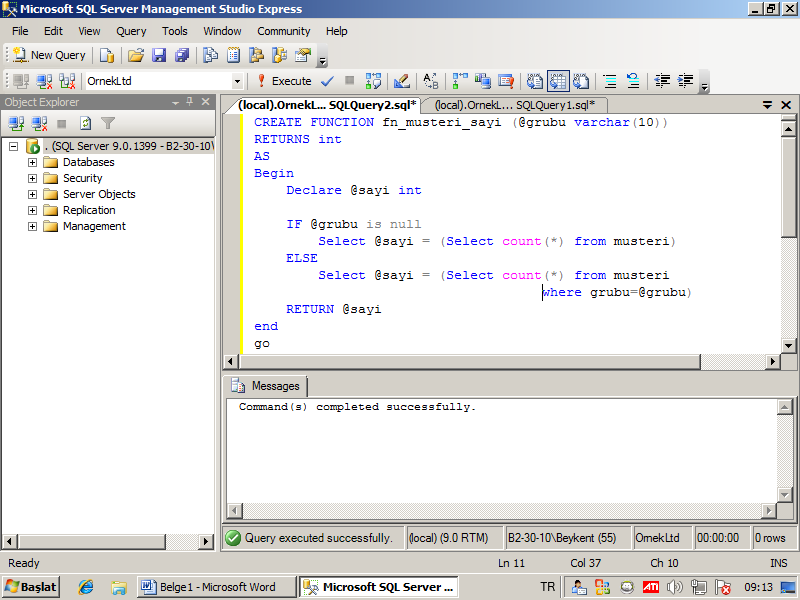
ELSE

Select @sayi = (Select count(\*) from musteri where grubu=@grubu)

RETURN @sayi

end

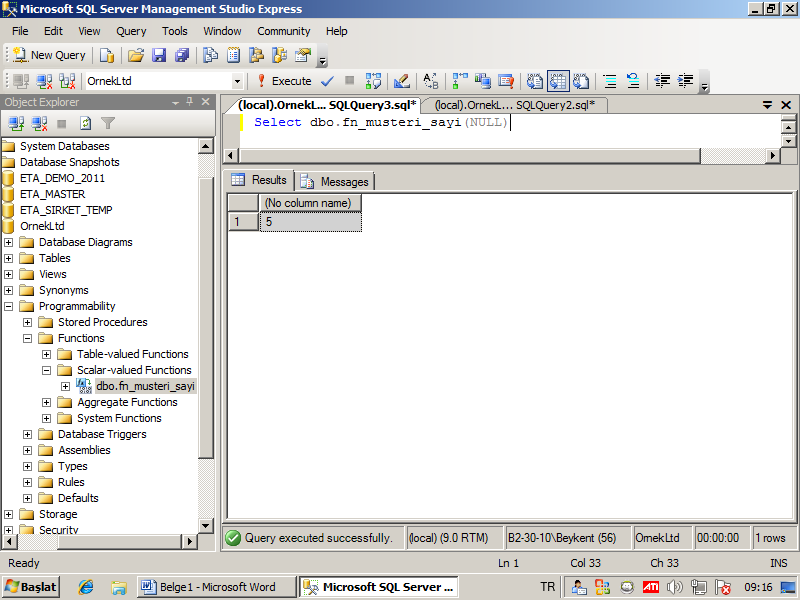
go



Yukarıdaki fonksiyonu çalıştıralım.

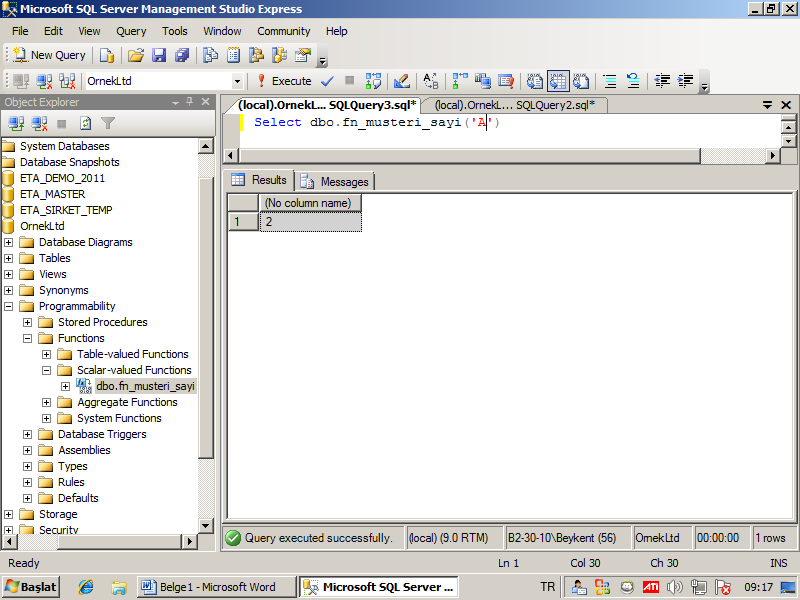
@grubu değeri olarak NULL girelim, bu bize tüm müşteri sayısını verecektir.

Select dbo.fn\_musteri\_sayi(NULL)



@grubu değeri olarak ‘A’ değerini girelim. Bu bize A grubundaki müşteri sayısını verecektir.

Select dbo.fn\_musteri\_sayi(‘A’)



TABLO DÖNDÜREN FONKSİYONLAR

(Table-Valued Functions)

Bu tip fonksiyonlar viewlere benzerdir. Bir tek select ifadesi içerir. View’den farklı olarak dışarıdan parametre alır.

Genel İfade

CREATE FUNCTION fonksiyonAdı (varsaParametreler)

RETURNS TABLE

AS

RETURN selectİfadesi

Go

View, fonksiyon ve prosedürün benzer ve farklılıklarını anlamak için aşağıdaki örnekleri inceleyelim.

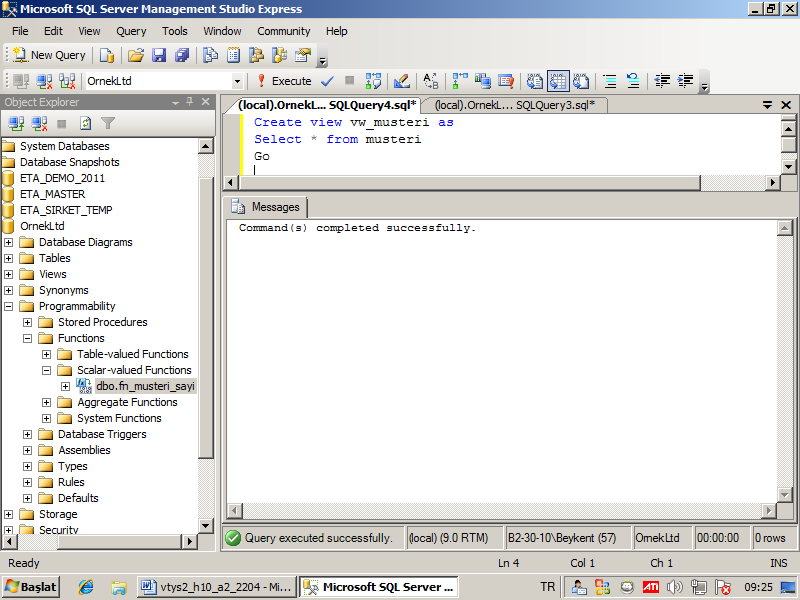
Örnek:

Tüm müşterileri listeleyen bir view oluşturunuz ve bu viewi sorgulayınız.

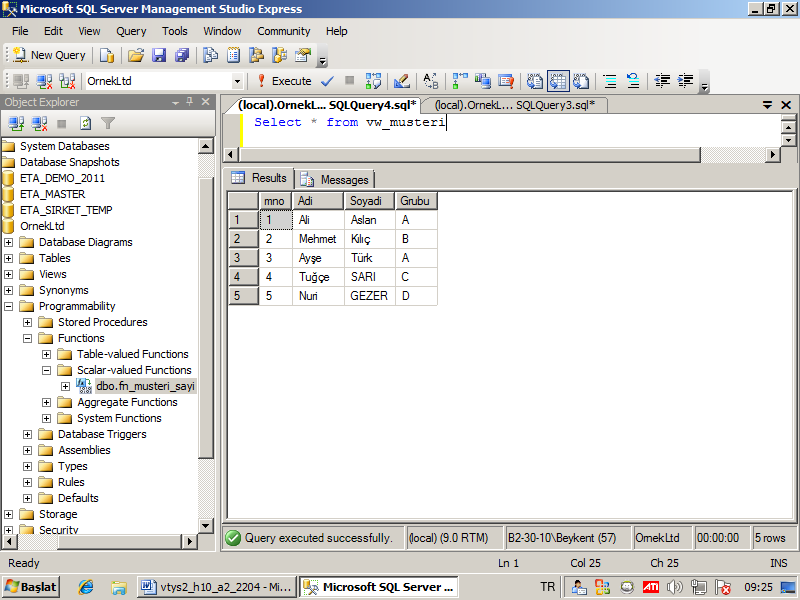
Create view vw\_musteri as

Select \* from musteri

Go



Select \* from vw\_musteri



Aynı işi yapan fonksiyon oluşturunuz.

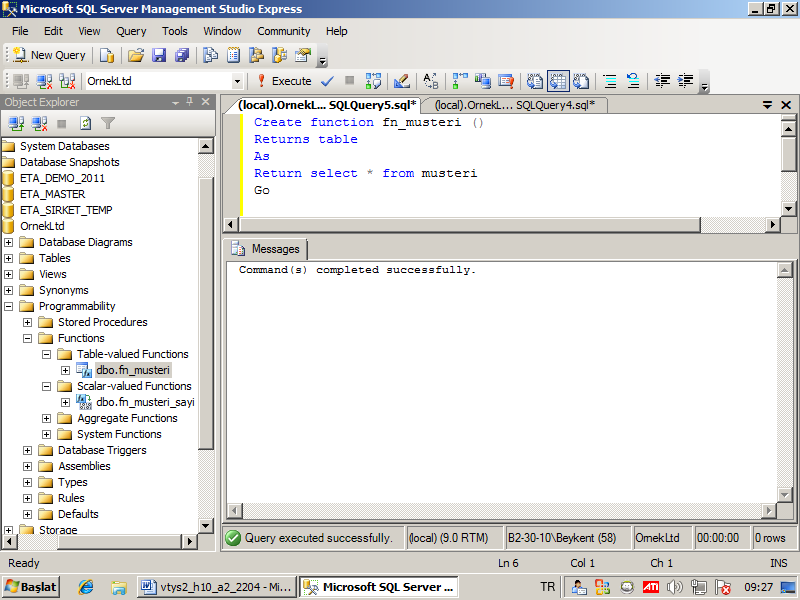
Create function fn\_musteri ()

Returns table

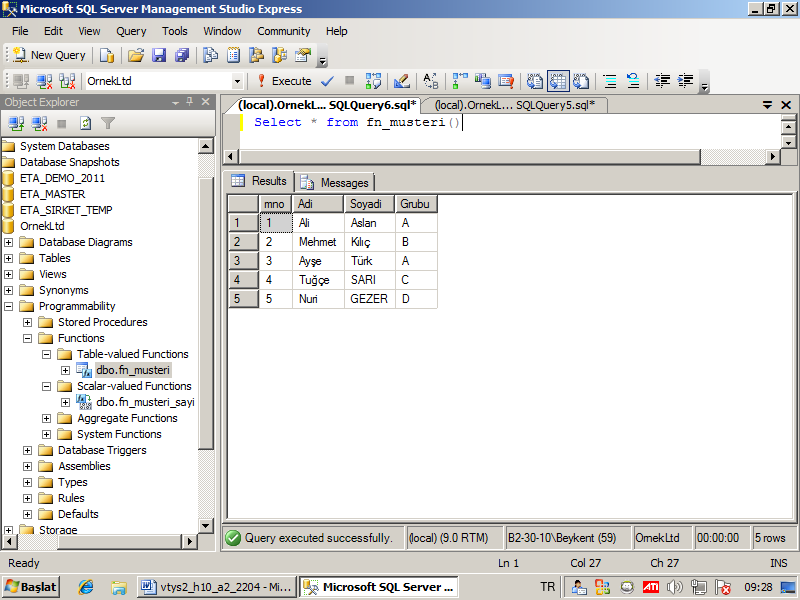
As

Return select \* from musteri

Go



Select \* from fn\_musteri()



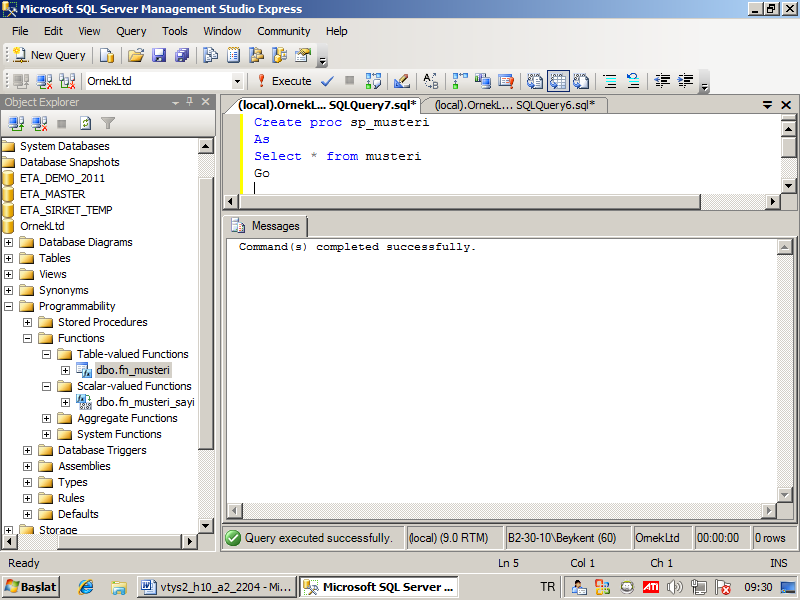
Yine aynı işi yapan prosedür oluşturalım.

Create proc sp\_musteri

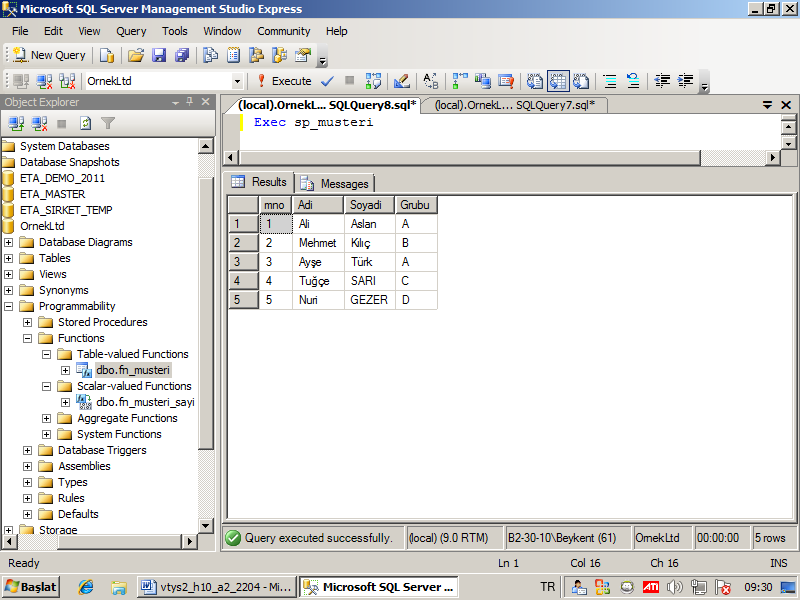
As

Select \* from musteri

Go



Exec sp\_musteri

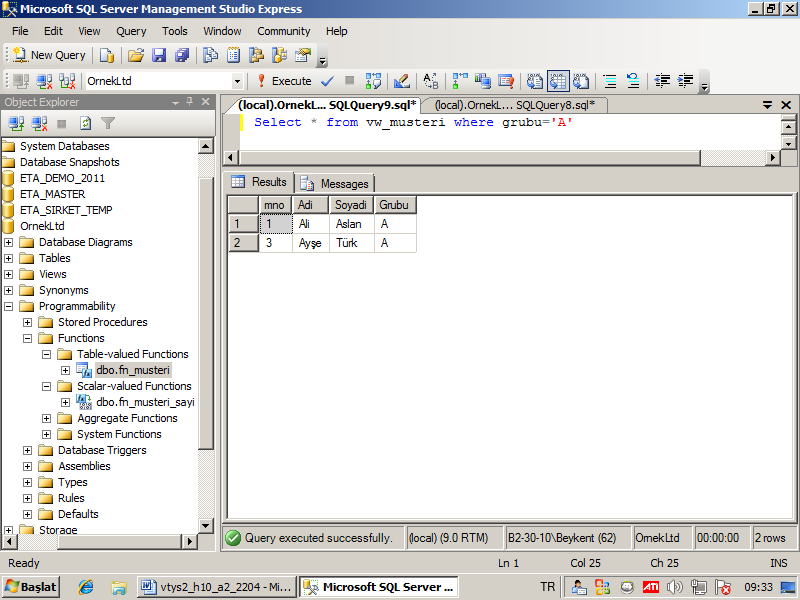


Yukarıdaki üç çıktıyı incelediğimizde aynı sonucu verdiğini görürüz.

Şimdi yukarıdaki view, fonksiyon ve prosedürden faydalanarak A grubundaki müşterileri listeleyelim.

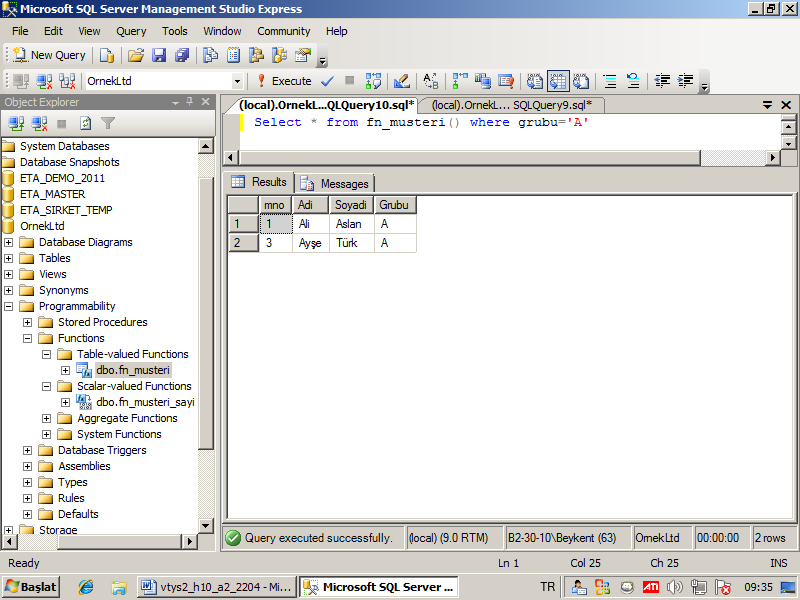
View’den faydalanarak A grubundakiler

Select \* from vw\_musteri where grubu=’A’



Fonksiyondan yararlanarak, A grubundakileri listeleyelim.

Select \* from fn\_muster() where grubu=’A’



Şimdi prosedürden A grubundakileri listeleyelim.

Sp\_musteri prosedüründen sadece tüm listeyi elde edebiliriz, A grubundakilere erişemeyiz. Prosedürler sorgulanamaz.

Yukarıdaki sorgulamalardan, fonksiyonların view gibi sorgulanabildiğini gördük.

Şimdi yukarıdaki view, fonksiyon ve prosedüre grubu parametresi tanımlayalım ve A grubundakileri listeleyelim.

View’e parametre tanımlayamayız.

Fonksiyona parametre tanımlayalım.

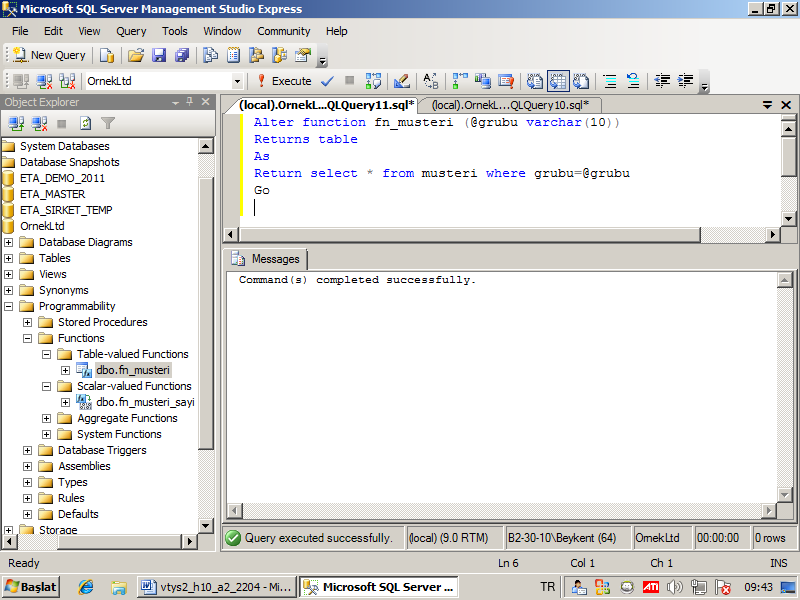
Alter function fn\_musteri (@grubu varchar(10))

Returns table

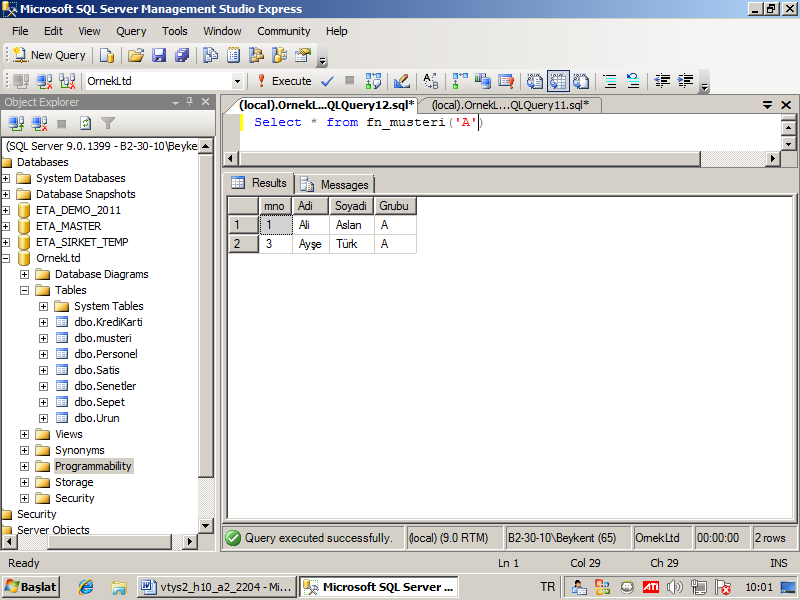
As

Return select \* from musteri where grubu=@grubu

Go



Select \* from fn\_musteri(‘A’)



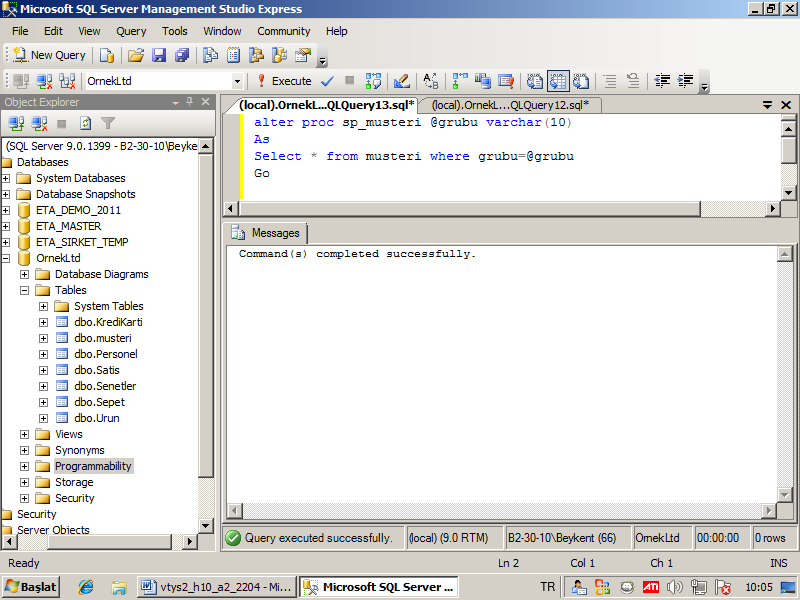
Aynı işi yapacak şekilde yukarıdaki prosedürü değiştirelim.

Alter proc sp\_musteri @grubu varchar(10)

As

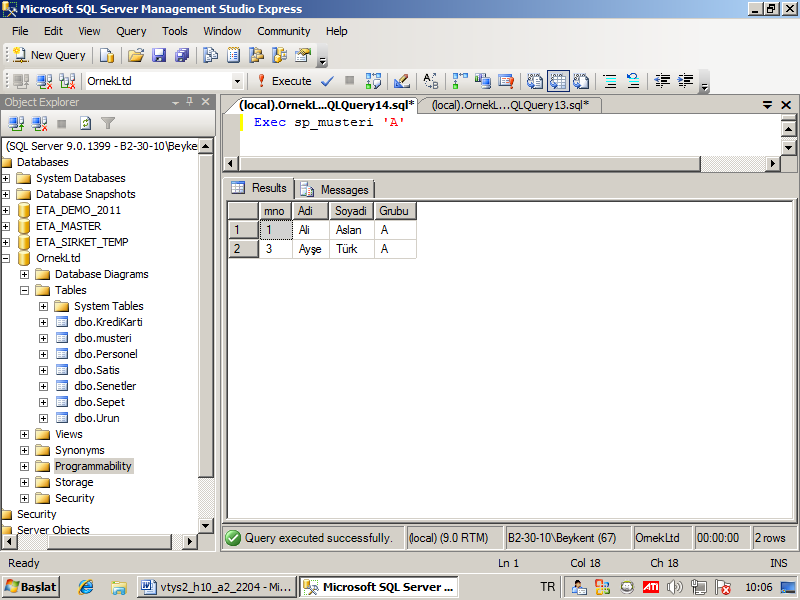
Select \* from musteri where grubu=@grubu

Go



Şimdi bu prosedürden A grubundaki müşterileri listeleyelim.

Exec sp\_musteri ‘A’



Şimdi yukarıdaki çıktılardan şu sonuca varırız.

1. view ve fonksiyon sorgulanabilir, prosedür sorgulanamaz.
2. View’in parametresi olamaz, ancak fonksiyon ve prosedürün parametresi bulunabilir.
3. Prosedürün kullanım amacı çoğunlukla, insert, update vs. gibi iş yapmaktır.

ÇOKLU İFADE İLE TABLO DÖNDÜREN FONKSİYONLAR

Bu türden fonksiyonlar bir öncekine benzerdir. Farkı dışarıya değer döndüren tablo yapısındaki bir değişkene fonksiyon içersinde muhtelif defa veri ekleme işlemi yapılabilir.

Genel ifade

CREATE FUNCTION fonkAdi (VarsaParametreler)

RETURNS @değer TABLE (Tablo Tanımı)

AS

BEGIN

Sql deyimi

insert into @değer selectİfadesi

insert into @değer selectİfadesi

insert into @değer selectİfadesi

…..

RETURN

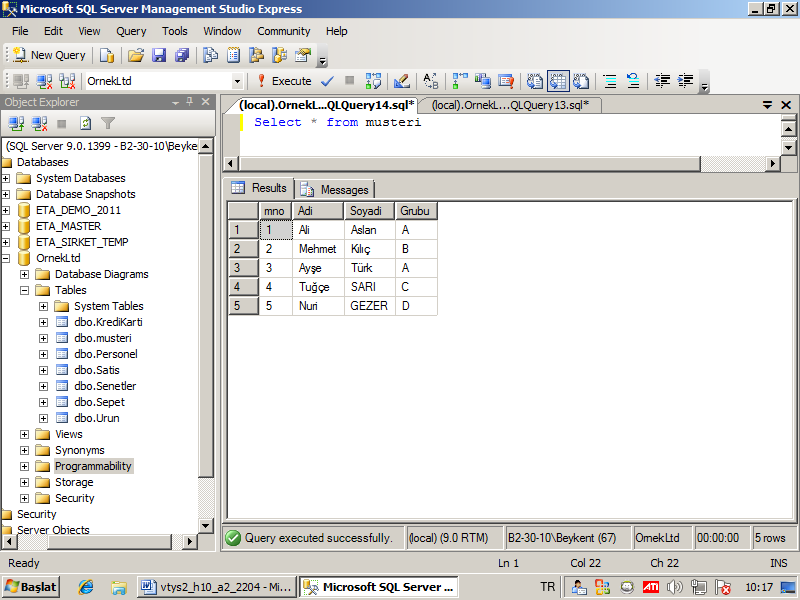
END

GO

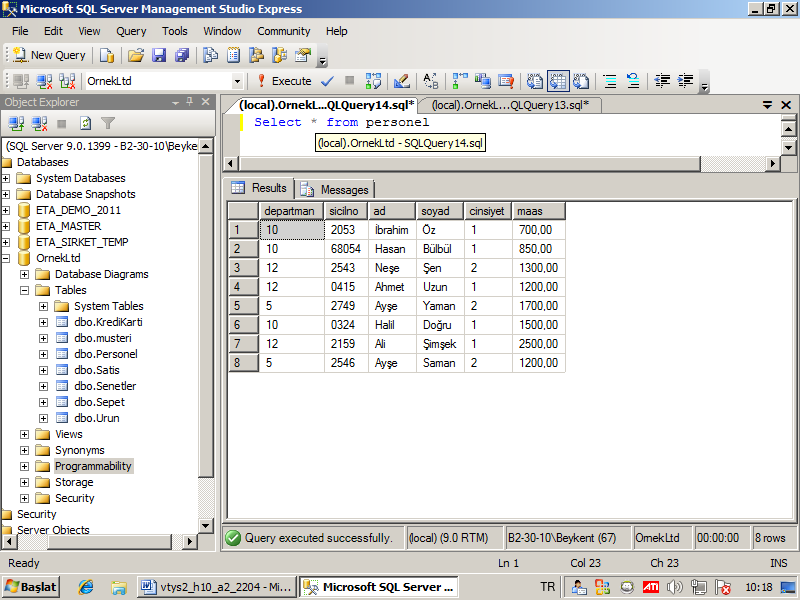
Örnek:

Müşteri ve Personeli aynı listede veren bir fonksiyon oluşturunuz.

Select \* from musteri



Select \* from personel



Create function fn\_per\_musteri()

RETURNS @deger TABLE (no varchar(10), ad varchar(10), soyad varchar(30), tipi char(1))

As

Begin

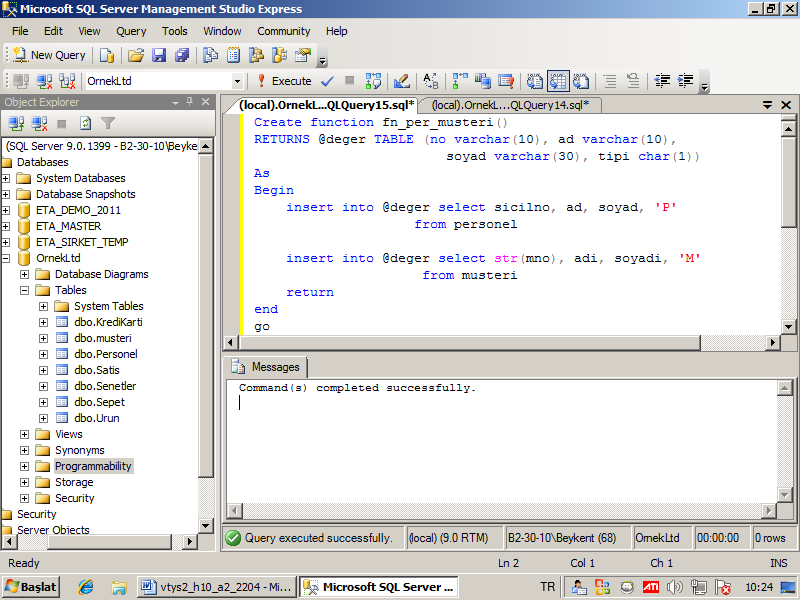
insert into @deger select sicilno, ad, soyad, ‘P’ from personel

insert into @deger select str(mno), adi, soyadi, ‘M’ from musteri

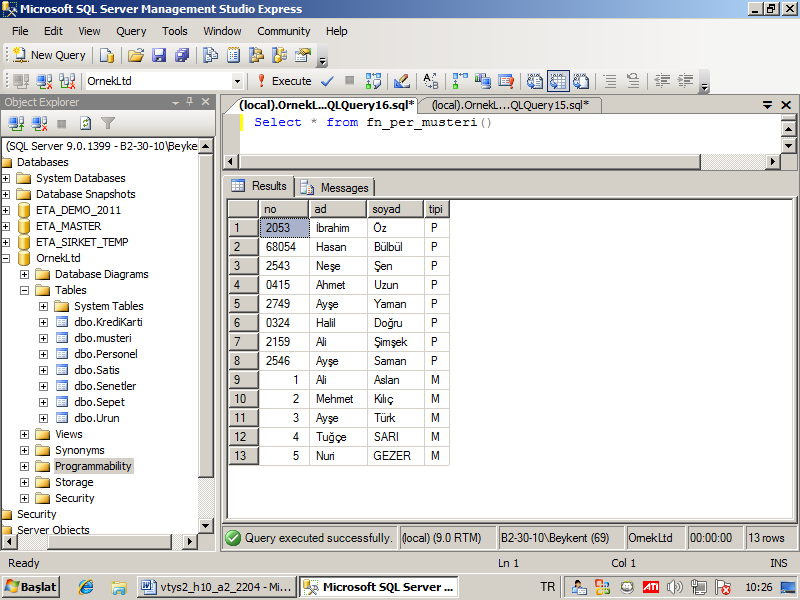
return

end

go



Select \* from fn\_per\_musteri()



Select \* from fn\_per\_musteri() where ad=’Ali’

