**--KULLANICI TANIMLI FONKSIYONLAR**

Skaler fonksiyonlar, tek bir değer döndüren fonksiyonlardır. Örnek getdate() bir scaler fonksiyondur. Çünkü bu fonksiyon tek bir değer döndürür yanı şu anın tarih ve saatini verir.

Genel oluşturma ifadesi

CREATE FUNCTION fonkAdı   
(varsaParametre )

RETURNS geriDonuşTipi

AS

BEGIN

SQL deyimleri

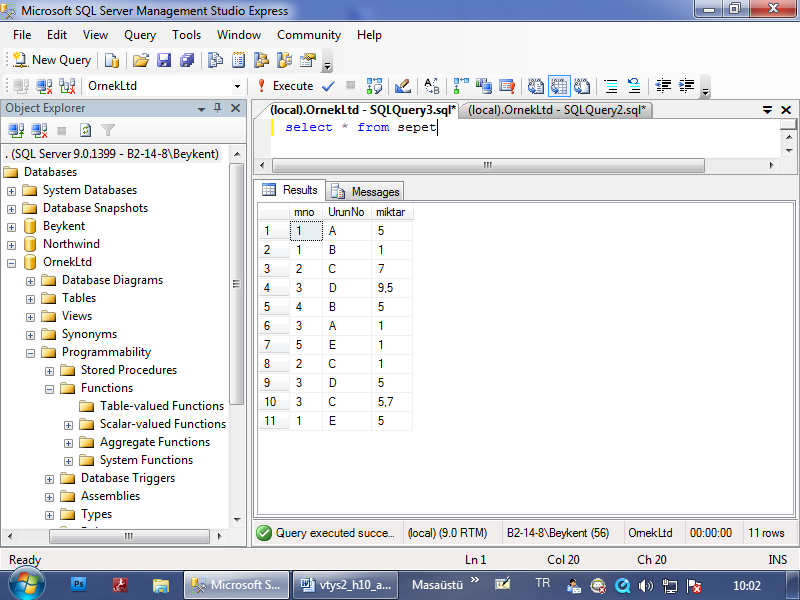
RETURN geriDönüşDeğeri

END

**Örnek:**

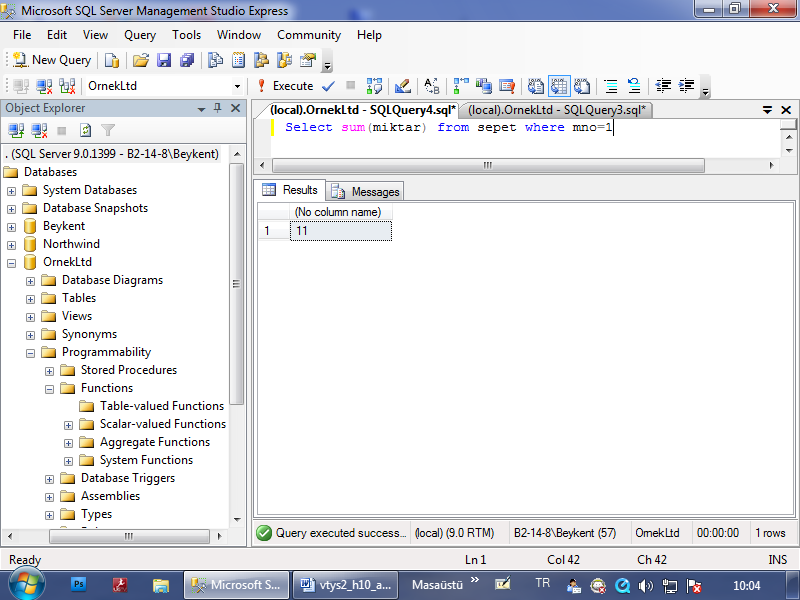
Bir müşteri no girildiğinde, bu müşterinin sepetinde toplam kaç ürün olduğunu bulan bir kullanıcı tanımlı fonksiyon oluşturunuz ve bu fonksiyonu çalıştırınız.

**Çözüm:**



Örneğin 1 nolu müşterinin miktar toplamını veren sorguyu yazalım.

Select sum(miktar) from sepet where mno=1



Yukarıdaki sorguyu fonksiyon içine koyalım.

CREATE FUNCTION fn\_urun\_sayi (@mno int) -- parametre

RETURNS float

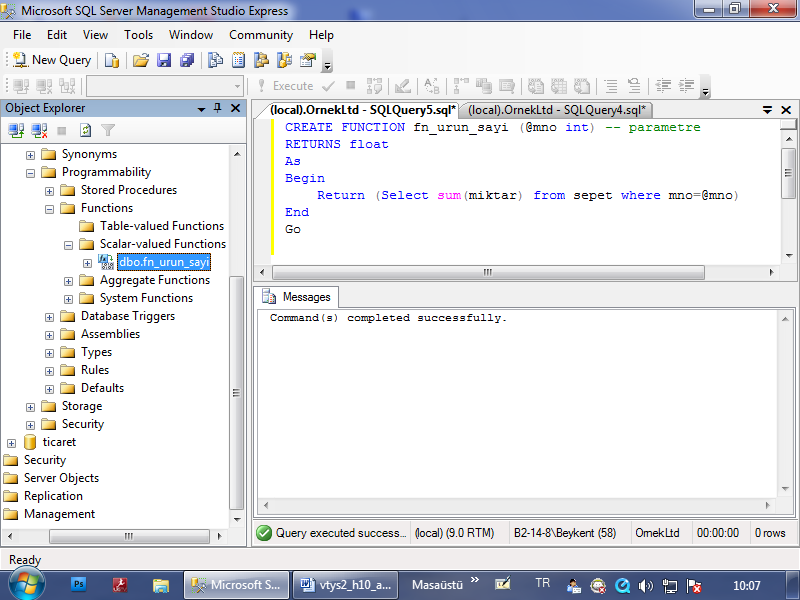
As

Begin

Return (Select sum(miktar) from sepet where mno=@mno)

End

Go

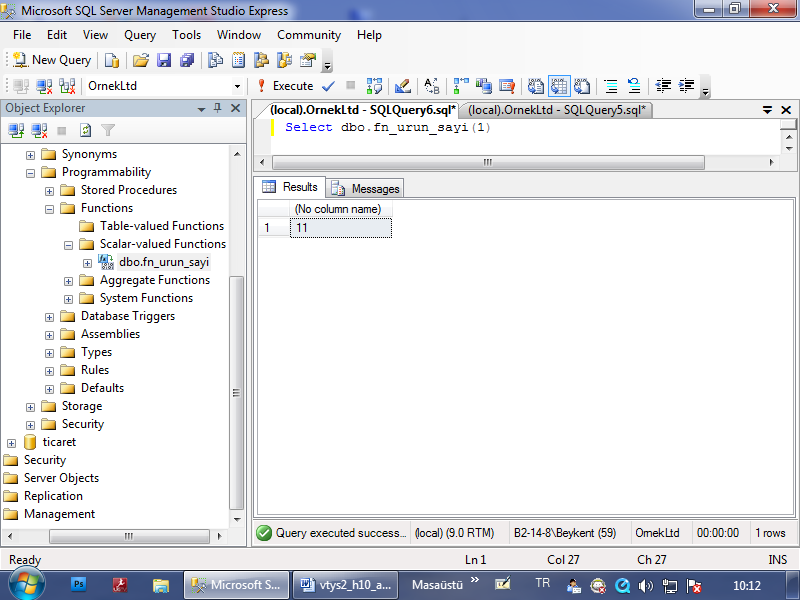


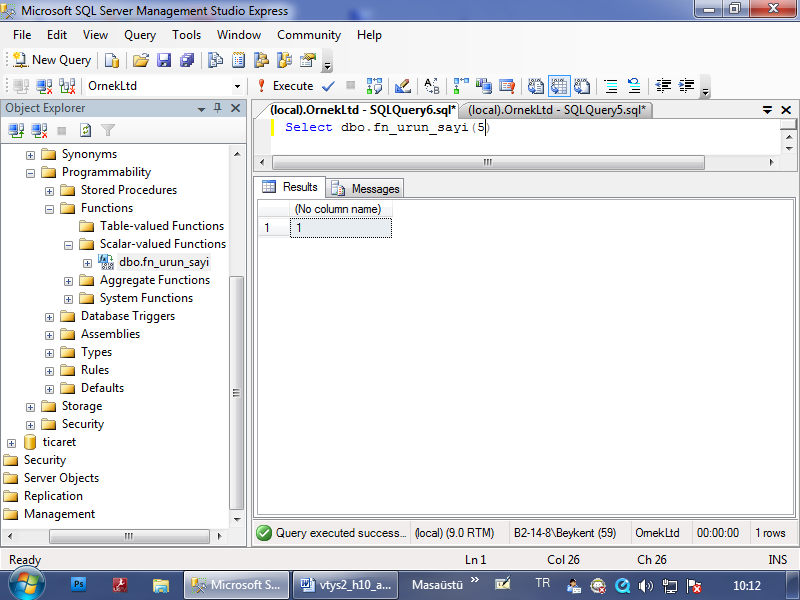
Yukarıdaki fonksiyonu çalıştıralım. Mesaja bakalım. Hatayı giderelim.

Select fn\_urun\_sayi(1)

Fonksiyon adı tanımlı değil mesajı. Biz kullanıcı tanımlı skaler fonksiyonu kullanırken şema adını da yazmalıyız.

Varsayılan şema dbo’dur. “select dbo.fn\_urun\_sayi(1)” şeklinde tanımlanmalıdır.





**Örnek:** 1,2 gibi ayın numarasını gireceğiz, Ocak, Şubat gibi ay adını döndüren bir fonksiyon oluşturunuz.

Create function fn\_ay\_adi (@ay tinyint)

Returns nvarchar(20)

As

Begin

Return (

Select case @ay

When 1 then ‘Ocak’

When 2 then ‘Şubat’

When 3 then ‘Mart’

When 4 then ‘Nisan’

When 5 then ‘Mayıs’

When 6 then ‘Haziran’

When 7 then ‘Temmuz’

When 8 then ‘Ağustos’

When 9 then ‘Eylül’

When 10 then ‘Ekim’

When 11 then ‘Kasım’

When 12 then ‘Aralık’

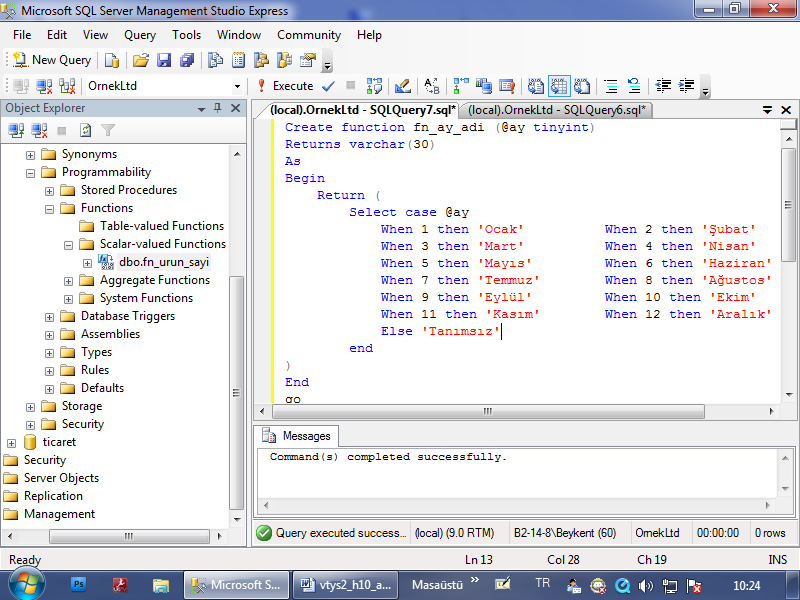
Else ‘Tanımsız’

end

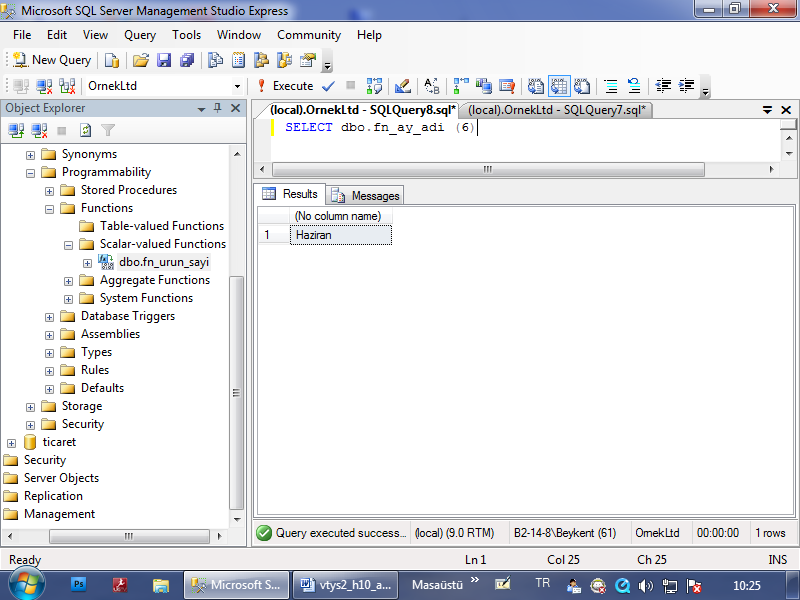
)

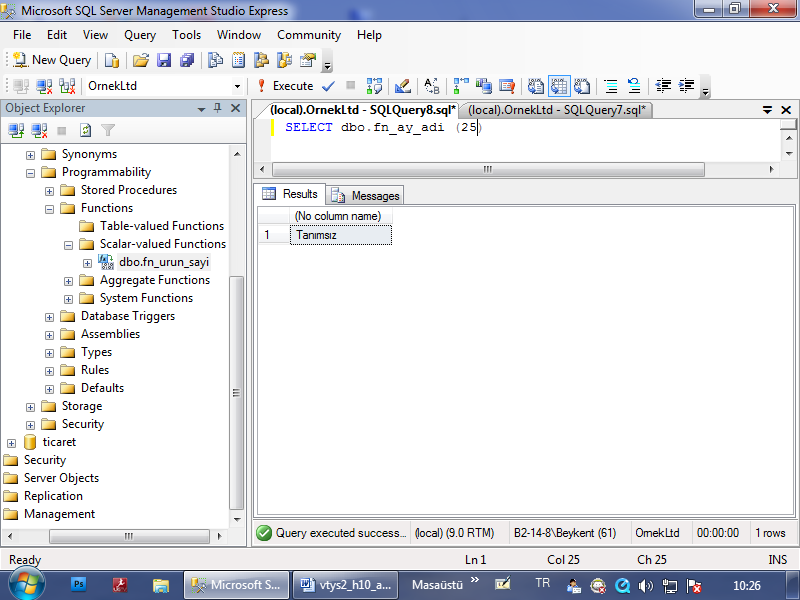
End

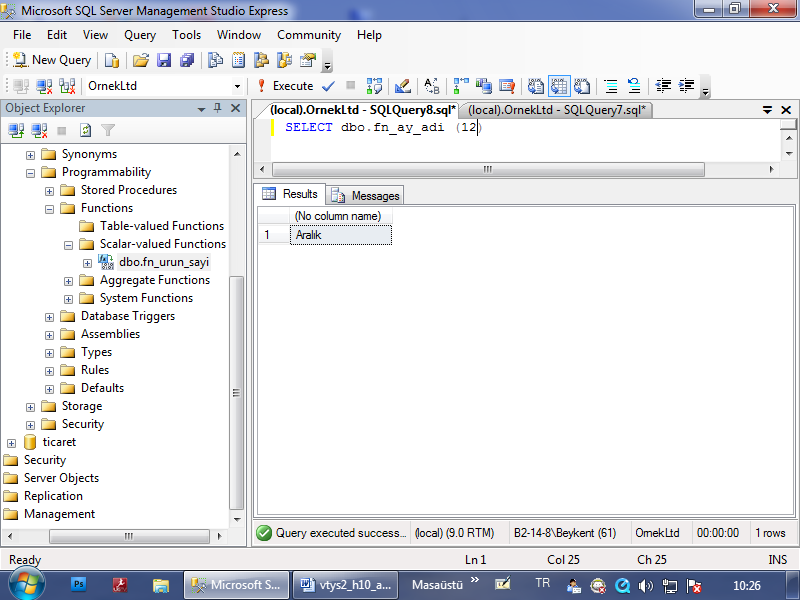
Go



SELECT dbo.fn\_ay\_adi (6)







**Örnek:**

Herhangi bir gruba ait müşteri sayısını veren bir fonksiyon oluşturunuz.

Create function fn\_musteri\_sayi (@grup varchar(10))

Returns int

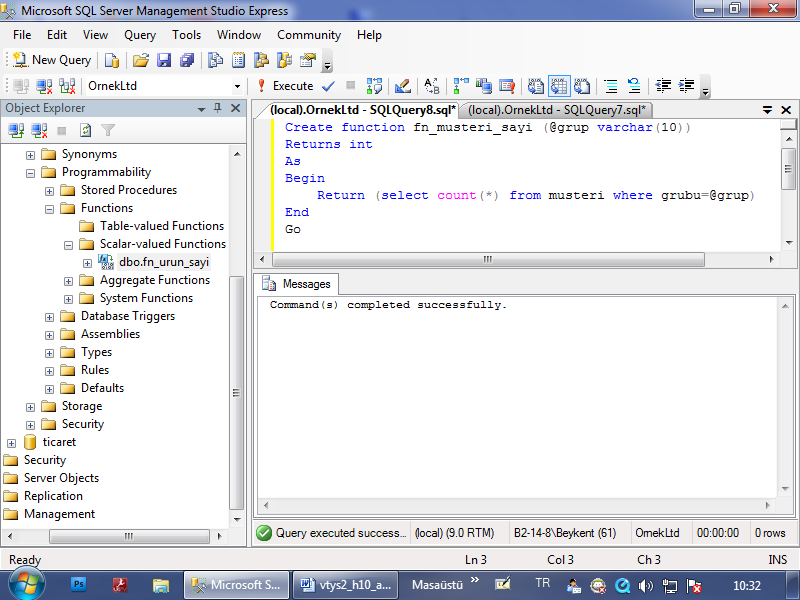
As

Begin

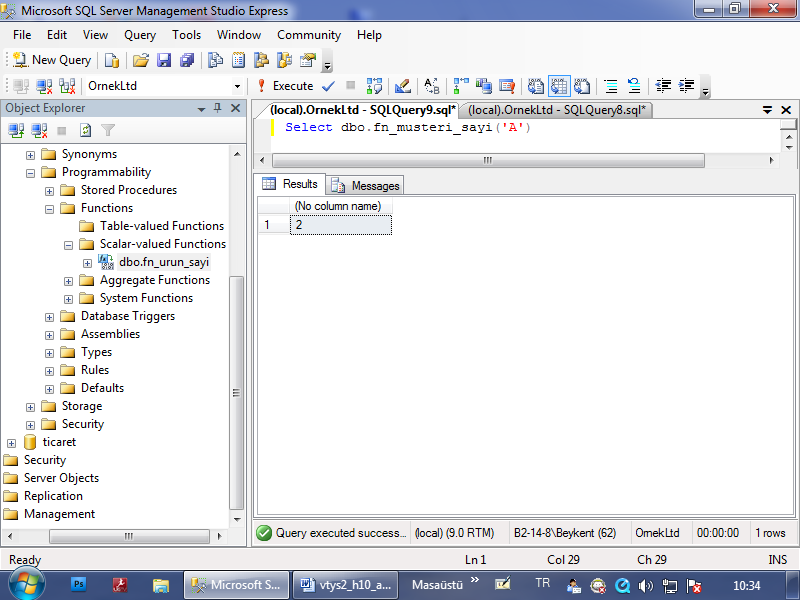
Return (select count(\*) from musteri where grubu=@grup)

End

Go



Select dbo.fn\_musteri\_sayi(‘A’)



Create function fn\_tum\_musteri\_sayi ()

Returns int

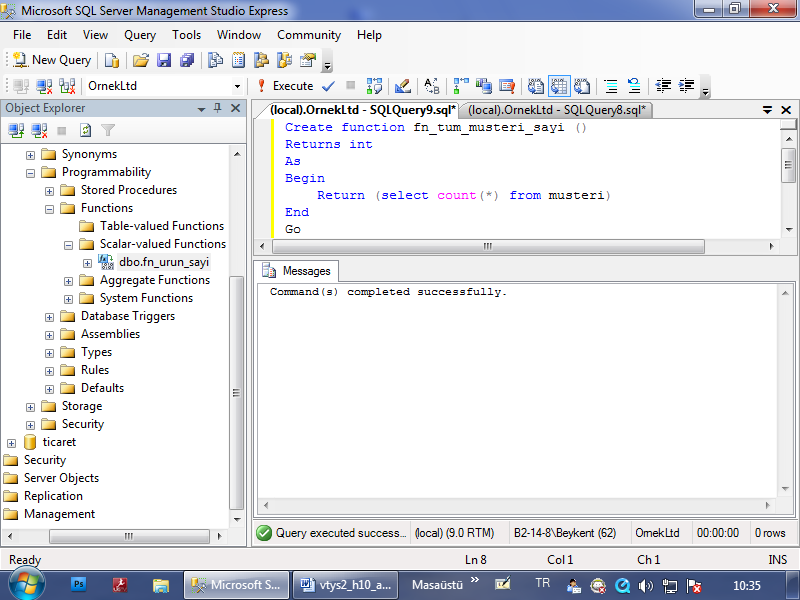
As

Begin

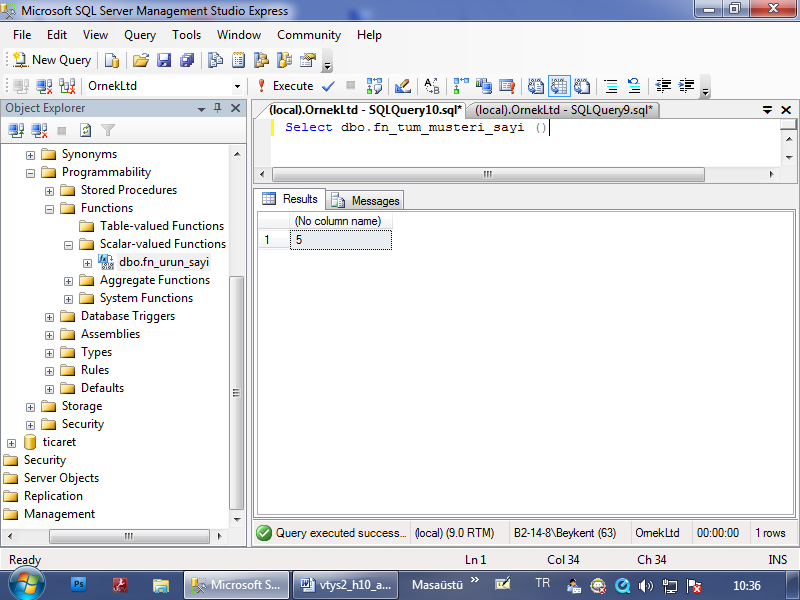
Return (select count(\*) from musteri)

End

Go



Select dbo.fn\_tum\_musteri\_sayi ()



**Örnek: ---** kisi\_id parametre olarak gönderilen kaydın adını ve soyadını birleştirerek geri döndüren fonksiyonu oluşturunuz.

CREATE FUNCTION tamisimal (@gelenkisi\_id int)

RETURNS varchar(100) AS

BEGIN

Declare @tam varchar(100)

Select @tam=ad+' '+soyad From isim

Where kisi\_id=@gelenkisi\_id

Return @tam

END

**Örnek: --**Veritabanının adres alnının boş olup olmadığını kontrol ederek adres girilmediyse ekrana adres girilmemiş yazdırın, eğer adres girilmişse girilen adres ekranda görünsün.

CREATE FUNCTION adreskontrol (@gelenadres varchar(50))

RETURNS varchar (50) AS

BEGIN

if @gelenadres is null

Set @gelenadres='Adres girilmemiş.'

Return @gelenadres

END

CREATE FUNCTION TOPLA(@X INT,@Y INT) RETURNS INT

AS

BEGIN

DECLARE @TOPLAM INT

SET @TOPLAM=@X+@Y

RETURN @TOPLAM

END

SELECT DBO.TOPLA(3,5)AS TOPLAM

SELECT \* FROM CALISANLAR

--YAS<25: GENÇ

--YAS>=25, YAS<45 : ORTAYAS

--YAS>45 : YASLI

CREATE FUNCTION DBO.DURUM(@X TINYINT) RETURNS VARCHAR(30)

AS

BEGIN

DECLARE @SONUC VARCHAR(30)

SET @SONUC= CASE WHEN @X<25 THEN 'GENÇ'

WHEN @X>=25 AND @X<40 THEN 'ORTA YAŞ'

ELSE 'YAŞLI'

END

RETURN @SONUC

END

SELECT SICILNO,ADSOYAD,YAS,DBO.DURUM(YAS) AS DURUM,SEHIR

FROM CALISANLAR

--DEĞİŞİKLİK VEYA SİLME İÇİB ALTER VEYA DROP FUNCTION FUNCTION ADI

**Fonksiyon Tanımlama**

--- kisi\_id parametre olarak gönderilen kaydın adını ve soyadını birleştirerek geri döndüren fonksiyonu oluşturunuz.

CREATE FUNCTION tamisimal (@gelenkisi\_id int)

RETURNS varchar(100) AS

BEGIN

Declare @tam varchar(100)

Select @tam=ad+' '+soyad From isim

Where kisi\_id=@gelenkisi\_id

Return @tam

END

--Veritabanının adres alanının boş olup olmadığını kontrol ederek adres girilmediyse ekrana adres girilmemiş yazdırın, eğer adres girilmişse girilen adres ekranda görünsün.

CREATE FUNCTION adreskontrol (@gelenadres varchar(50))

RETURNS varchar (50) AS

BEGIN

if @gelenadres is null

Set @gelenadres='Adres girilmemiş.'

Return @gelenadres

END