STORED PROCEDURE

Birden fazla işlemi, paketlenmiş bir halde bir tek komut ile çalıştırmamız gerektiğinde stored prosedürler kullanılır.

Prosedürlerin içinde her türlü Transact-SQL deyimi bulunabilir.

Ağ bazlı uygulamalarda, ağ trafiğini azaltır.

Genel ifade:

CREATE: oluşturmak

ALTER : nesne yapısını değiştirmek

DROP : nesneyi silmek

CREATE PROC

Veya

CREATE PROCEDURE

CREAT PROC prosedürAdı

varsaParametreler

AS

….

Sql deyimleri

….

Go

Örnek:

İki hesap arasında para transferi yapan bir prosedür oluşturunuz. Bakiye sıfırın altına inerse “bakiye yetersiz” mesajı versin, sıfırın altına düşmüyorsa para transferi gerçekleşsin.

Kullanacağım parametreler

Gönderici hesabın nosu

Alıcı hesabın nosu

Gönderilen para miktarı

CREATE PROC sp\_para\_gonder

@gonderen varchar(10),

@alici varchar(10),

@para Money

AS

….

Para gönderen kodu yaz.

….

Go

IF (gönderenin bakiyesi)>=@para

Parayı gönder

ELSE

“bakiye yetersiz” mesajı ver.

IF (select bakiye from hesaplar where hesapno=@gonderen)>=@para

-- gönderenin parasını azalt

Begin

Update hesaplar set bakiye=bakiye-@para where hesapno=@gonderen

-- alıcının parasını artır

Update hesaplar set bakiye=bakiye+@para where hesapno=@alici

End

ELSE

Print ‘bakiye yetersiz’

Yukarıdaki iki kodu birleştirelim

CREATE PROC sp\_para\_gonder

@gonderen varchar(10),

@alici varchar(10),

@para Money

AS

IF (select bakiye from hesaplar where hesapno=@gonderen)>=@para

Begin

Update hesaplar set bakiye=bakiye-@para where hesapno=@gonderen

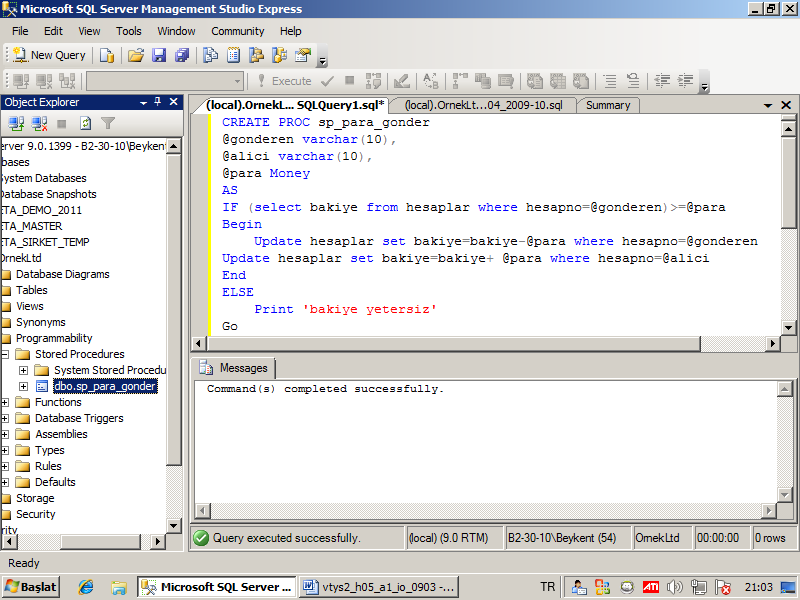
Update hesaplar set bakiye=bakiye+ @para where hesapno=@alici

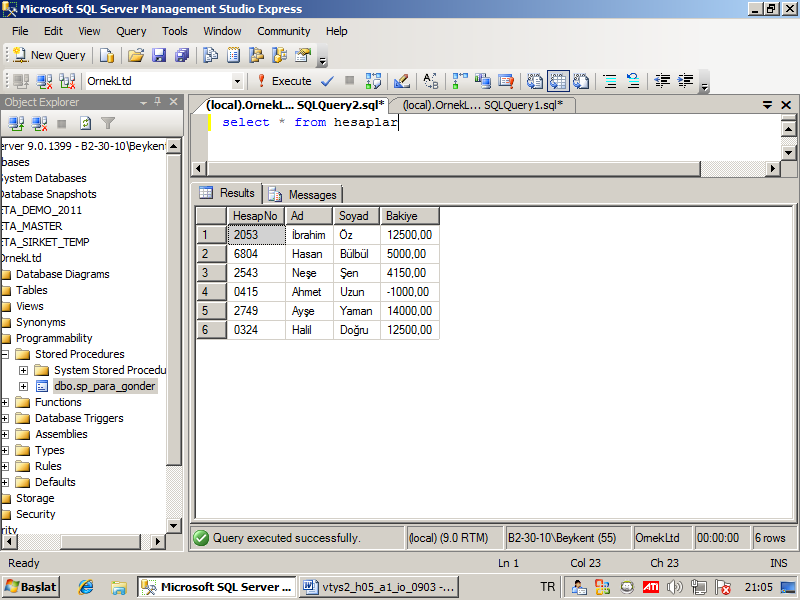
End

ELSE

Print ‘bakiye yetersiz’

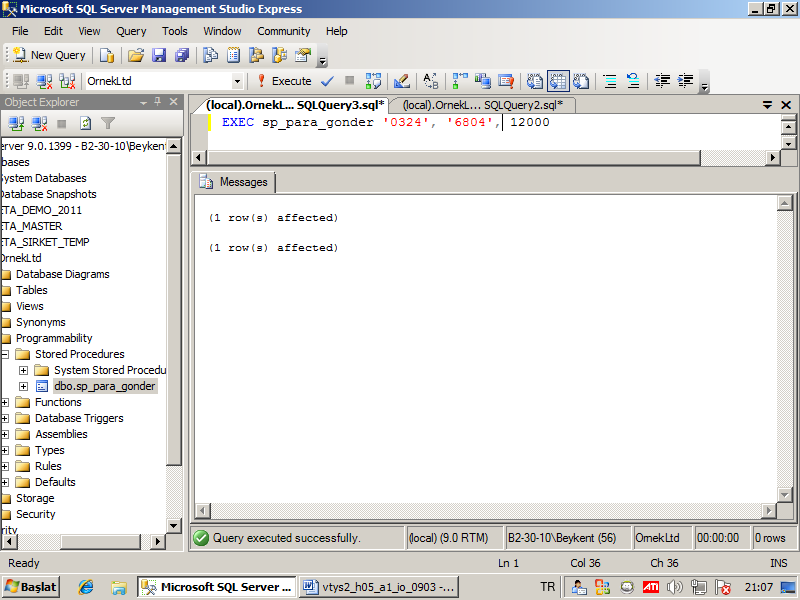
Go

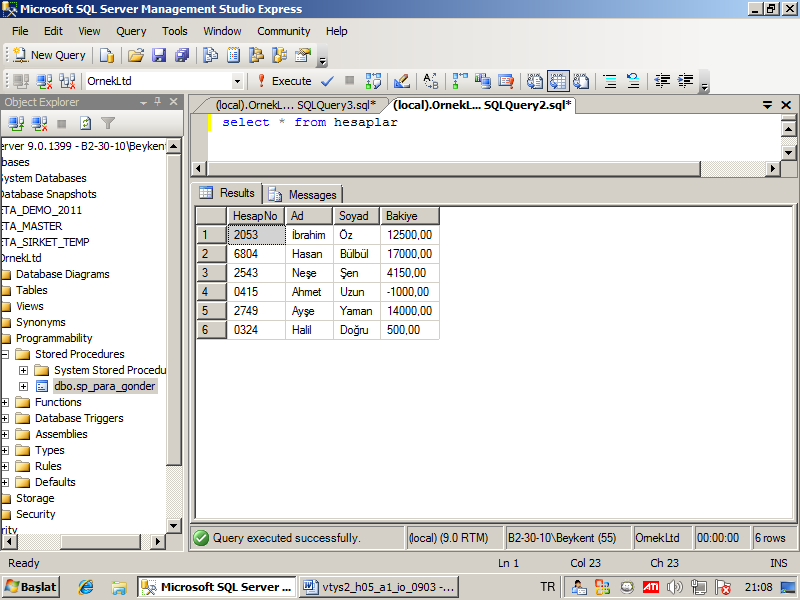




Yukarıdaki prosedürü çalıştırarak 0324 nolu hesaptan 6804 nolu hesaba 12000 değerini gönderiniz.

EXEC sp\_para\_gonder ‘0324’, ‘6804’, 12000

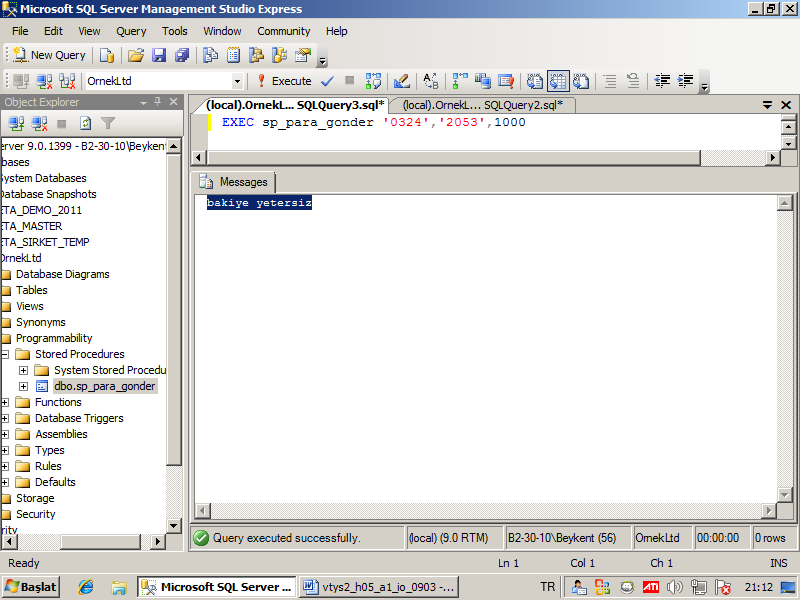




Örnek:

Yukarıdaki prosedürü kullanarak 0324 nolu hesaptan 2053 nolu hesaba 1000 değerini gönderiniz.

EXEC sp\_para\_gonder ‘0324’,’2053’,1000



Yukarıdaki EXEC deyimi parametre isimleri kullanarak yazılabilir. Bu durumda parametre sıralaması önemli değildir.

EXEC sp\_para\_gonder @gonderen=‘0324’, @alici=’2053’, @para=1000

Veya

EXEC sp\_para\_gonder @para=1000, @gonderen=‘0324’, @alici=’2053’

Parametreye varsayılan değer atayalım

@parametre veritipi = değer

Yukarıdaki prosedürdeki parametreye varsayılan değer atayalım.

ALTER PROC sp\_para\_gonder

@gonderen varchar(10)=NULL,

@alici varchar(10)=NULL,

@para Money=0

AS

IF (select bakiye from hesaplar where hesapno=@gonderen)>=@para

Begin

Update hesaplar set bakiye=bakiye-@para where hesapno=@gonderen

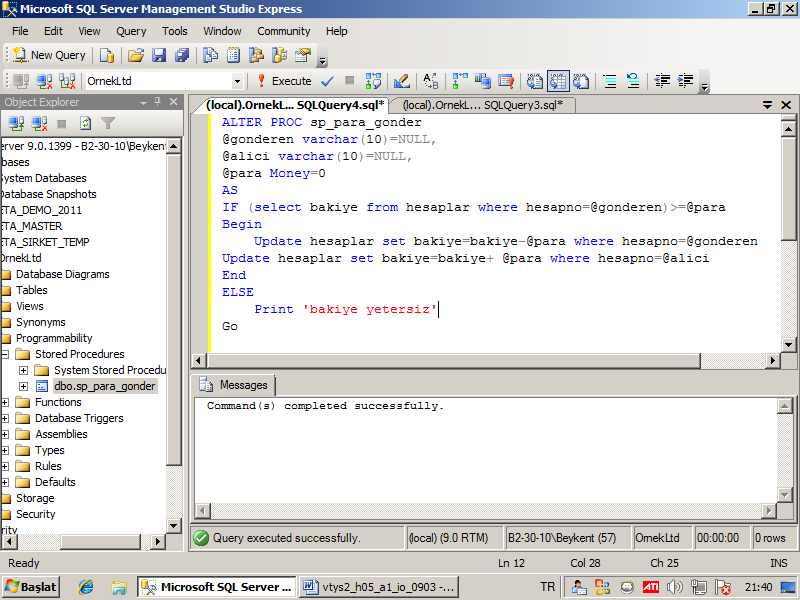
Update hesaplar set bakiye=bakiye+ @para where hesapno=@alici

End

ELSE

Print ‘bakiye yetersiz’

Go



EXEC sp\_para\_gonder

EXEC sp\_para\_gonder ‘0324’

Örnek:

Verilen bir hesap numarasının bakiyesini artırıp azaltan bir prosedür yazınız. Kullanacağımız parametreler, hesap\_no

,para miktarı ve işlem kodu (işlem kodu 1 azalt değilse artır)

CREATE PROC sp\_bakiye\_artir\_azalt

@hesap\_no varchar(10),

@para Money,

@islem\_kodu tinyint=1

AS

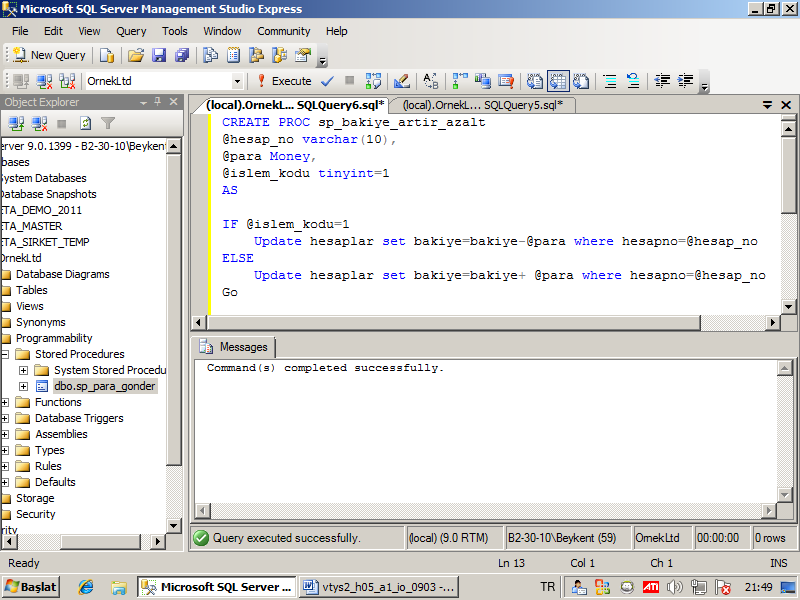
IF @islem\_kodu=1

Update hesaplar set bakiye=bakiye-@para where hesapno=@hesap\_no

ELSE

Update hesaplar set bakiye=bakiye+ @para where hesapno=@hesap\_no

Go

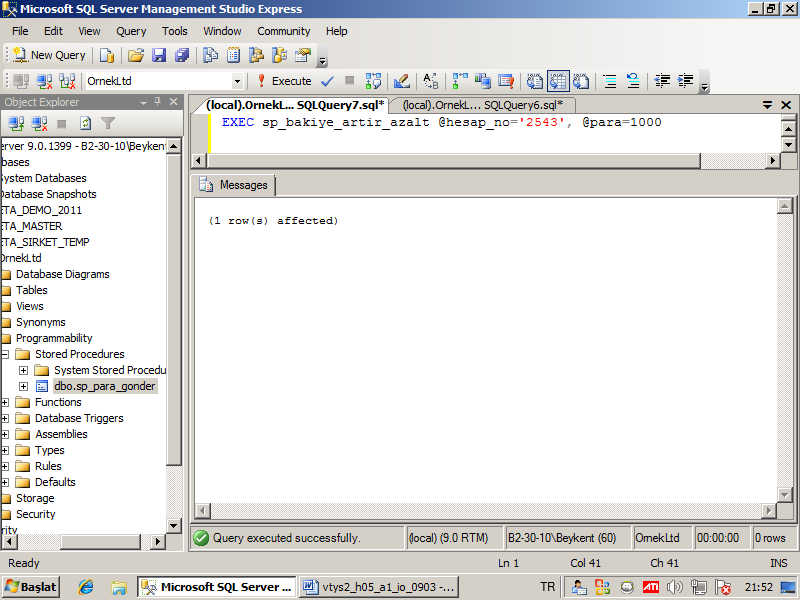


2543 nolu hesabın bakiyesini 1000 azalt

EXEC sp\_bakiye\_artir\_azalt @hesap\_no=’2443’, @para=1000, @islem\_kodu=1

Veya

EXEC sp\_bakiye\_artir\_azalt @hesap\_no=’2543’, @para=1000



2543 nolu hesabın bakiyesini 1000 artır

EXEC sp\_bakiye\_artir\_azalt @hesap\_no=’2543’, @para=1000, @islem\_kodu=0

