

## Örnek:

Okul numarası verilen öğrencinin notlarını ve ders adlarını listeleyen fonksiyon oluşturunuz.

## Çözüm:

```
B2-30-HOCA\SQ...QLQuery1.sql* B2-30-HC
SELECT * FROM Notlar
```

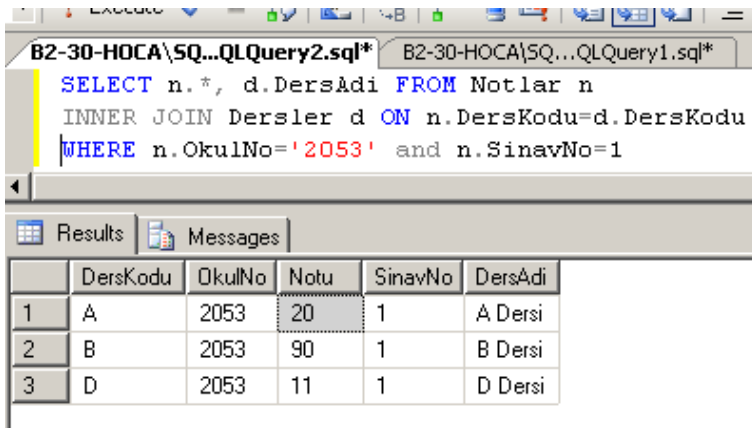
	DersKodu	OkulNo	Notu	SinavNo
1	A	2053	20	1
2	B	2053	90	1
3	D	2053	11	1
4	C	6805	0	1

```
B2-30-HOCA\SQ...QLQuery1.sql* B2
SELECT * FROM Dersler
```

	DersKodu	DersAdi	Kredisi
1	A	A Dersi	2
2	B	B Dersi	4
3	C	C Dersi	3

Fonksiyonu yazmadan önce, bu fonksiyon içinde bulunacak sorguyu oluşturalım. Örnek olarak 2053 nolu öğrencinin 1 nolu sınavda aldığı notları listeleyelim.

```
SELECT n.*, d.DersAdi FROM Notlar n
INNER JOIN Dersler d ON
n.DersKodu=d.DersKodu WHERE
n.OkulNo='2053' and n.SinavNo=1
```



The screenshot shows a SQL query execution window with two tabs: 'B2-30-HOCA\SQL...QLQuery2.sql\*' and 'B2-30-HOCA\SQL...QLQuery1.sql\*'. The query in the active tab is: `SELECT n.*, d.DersAdi FROM Notlar n INNER JOIN Dersler d ON n.DersKodu=d.DersKodu WHERE n.OkulNo='2053' and n.SinavNo=1`. Below the query, there are tabs for 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with the following data:

	DersKodu	OkulNo	Notu	SinavNo	DersAdi
1	A	2053	20	1	A Dersi
2	B	2053	90	1	B Dersi
3	D	2053	11	1	D Dersi

Yukarı sorguda 2 değişken var biri OkulNo diğer sınavNo. Bunların parametre olduğu fonksiyonu yazalım.

```
CREATE FUNCTION fnOgrenciNotlar
(@OkulNo char(10), @SinavNo tinyint)
RETURNS TABLE
```

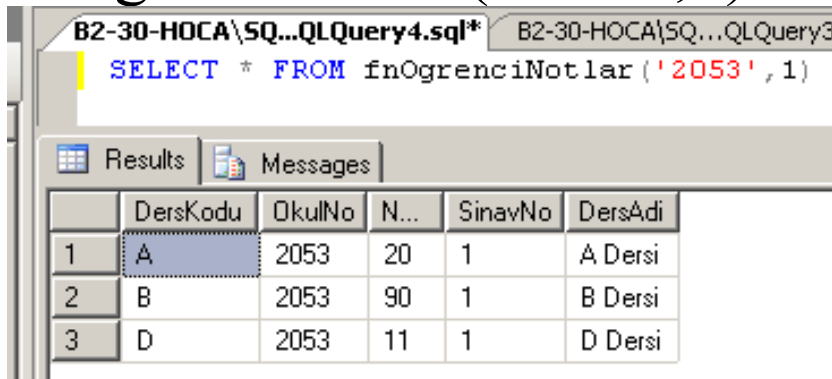
AS

```
RETURN SELECT n.*, d.DersAdi FROM  
Notlar n INNER JOIN Dersler d ON  
n.DersKodu=d.DersKodu WHERE  
n.OkulNo=@OkulNo and  
n.SinavNo=@SinavNo
```

go

Yukarıdaki fonksiyondan faydalanarak 2053 nolu öğrencinin 1 nolu sınavdan aldığı notları listeleyiniz.

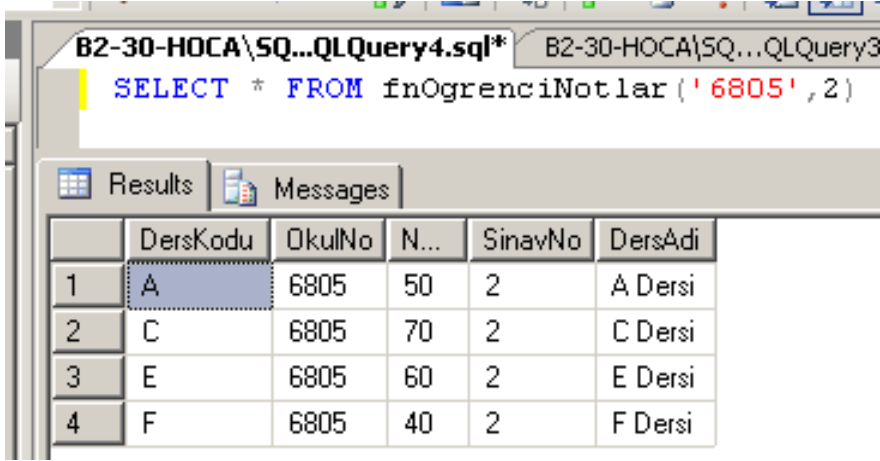
```
SELECT * FROM  
fnOgrenciNotlar('2053',1)
```



	DersKodu	OkulNo	N...	SinavNo	DersAdi
1	A	2053	20	1	A Dersi
2	B	2053	90	1	B Dersi
3	D	2053	11	1	D Dersi

-- 6805 nolu öğrencinin 2 nolu sınavdan aldığı notlar

```
SELECT * FROM  
fnOgrenciNotlar('6805',2)
```



The screenshot shows a SQL query window with the following query: `SELECT * FROM fnOgrenciNotlar ('6805', 2)`. The results are displayed in a table with the following columns: DersKodu, OkulNo, N..., SinavNo, and DersAdi. The results are as follows:

	DersKodu	OkulNo	N...	SinavNo	DersAdi
1	A	6805	50	2	A Dersi
2	C	6805	70	2	C Dersi
3	E	6805	60	2	E Dersi
4	F	6805	40	2	F Dersi

Örnek: Herhangi bir dersin sınav ortalamasını veren fonksiyon yazınız.

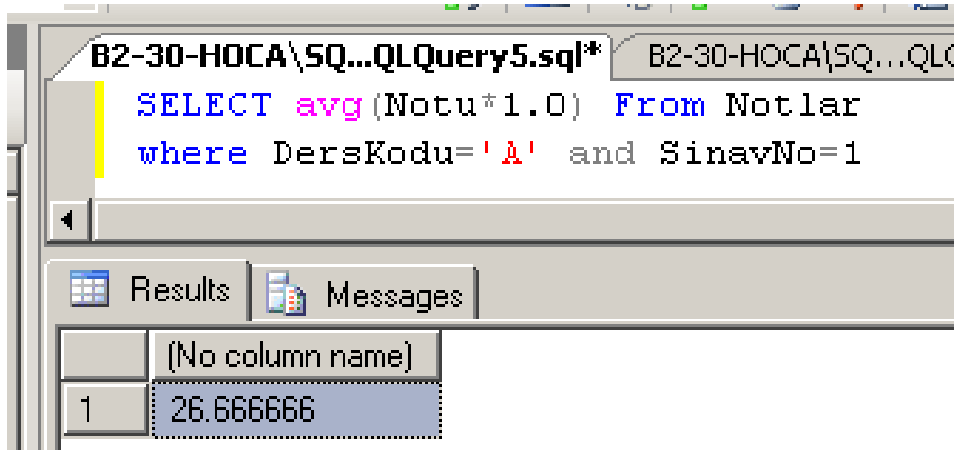
Çözüm:

Örneđin A kodlu dersin 1 nolu sınavının ortalamasını bulalım.

`SELECT avg(Notu*1.0) From Notlar where DersKodu='A' and SinavNo=1`

$$(1+2)/2 = 1$$

$$(1.0+2.0)/2 = 1.5$$

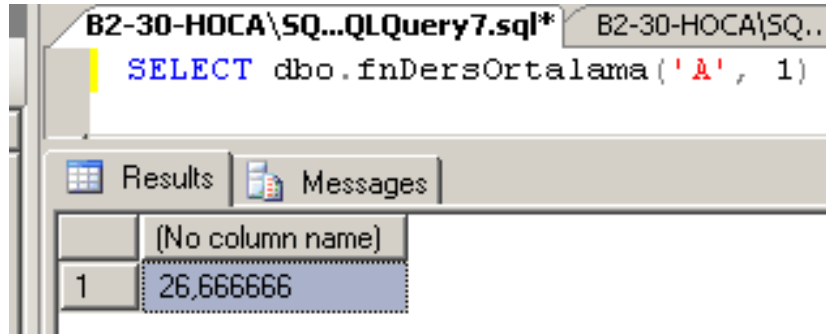


Yukarıdaki kodu fonksiyon içine yerleştirelim.

```
CREATE FUNCTION fnDersOrtalama
(@DersKodu char(10), @SinavNo tinyint)
RETURNS float
AS
BEGIN
    RETURN
    ( SELECT avg(Notu*1.0) FROM
Notlar WHERE DersKodu=@DersKodu
and SinavNo=@SinavNo)
END
go
```

A kodlu dersin 1 nolu sınavının ortalaması

SELECT dbo.fnDersOrtalama('A', 1)



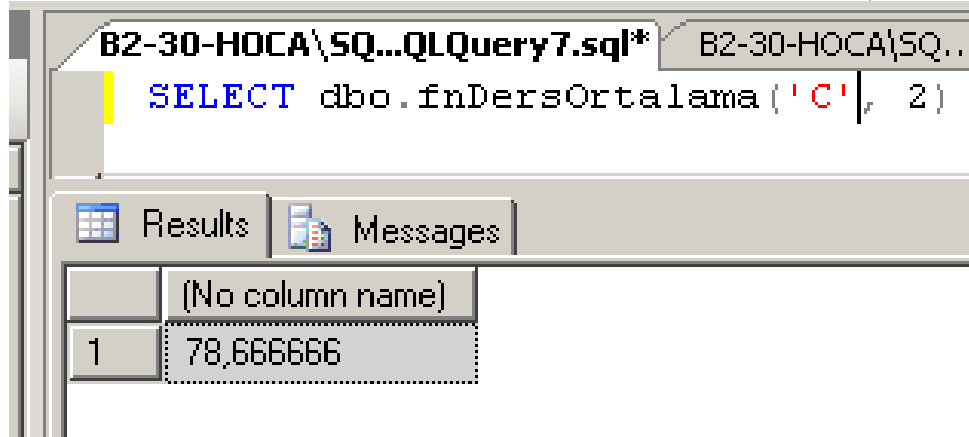
B2-30-HOCA\SQ...QLQuery7.sql\* B2-30-HOCA\SQ...

```
SELECT dbo.fnDersOrtalama('A', 1)
```

Results Messages

	(No column name)
1	26,666666

C kodlu dersin 2 nolu sınavının ortalaması  
SELECT dbo.fnDersOrtalama('C', 2)



B2-30-HOCA\SQ...QLQuery7.sql\* B2-30-HOCA\SQ...

```
SELECT dbo.fnDersOrtalama('C', 2)
```

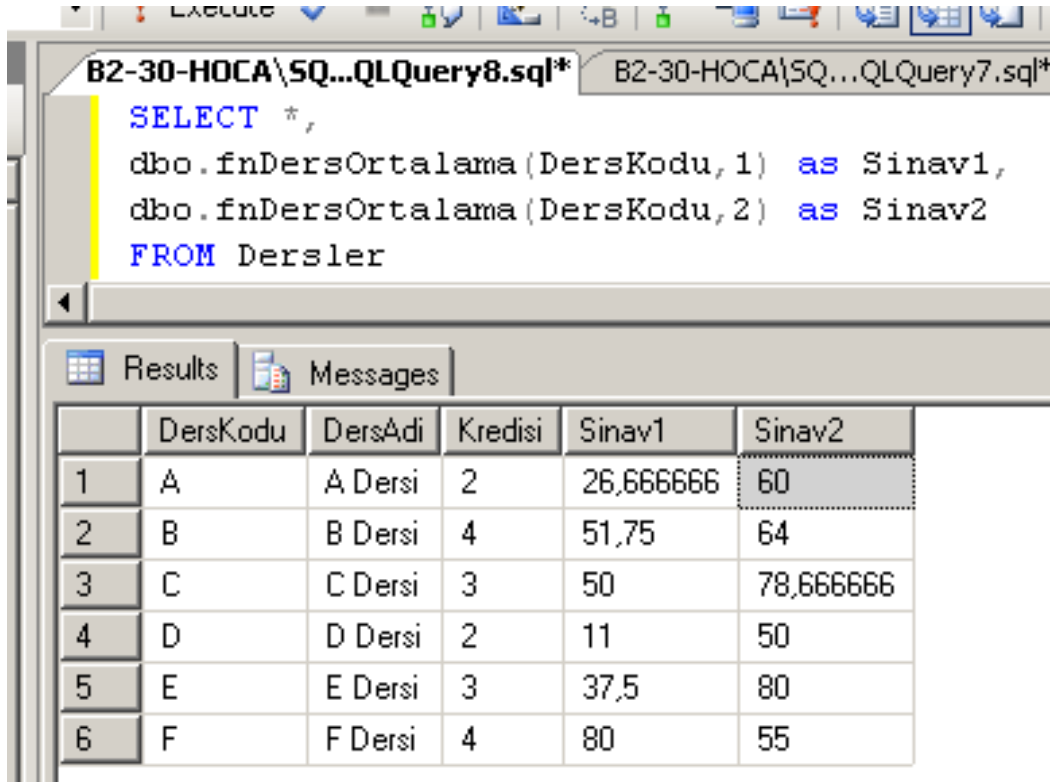
Results Messages

	(No column name)
1	78,666666

Örnek:

Dersler tablosunu ortalamaları ile birlikte listeleyen sorguyu yazınız.

```
SELECT *,  
dbo.fnDersOrtalama(DersKodu,1) as  
Sinav1, dbo.fnDersOrtalama(DersKodu,2)  
as Sinav2 FROM Dersler
```



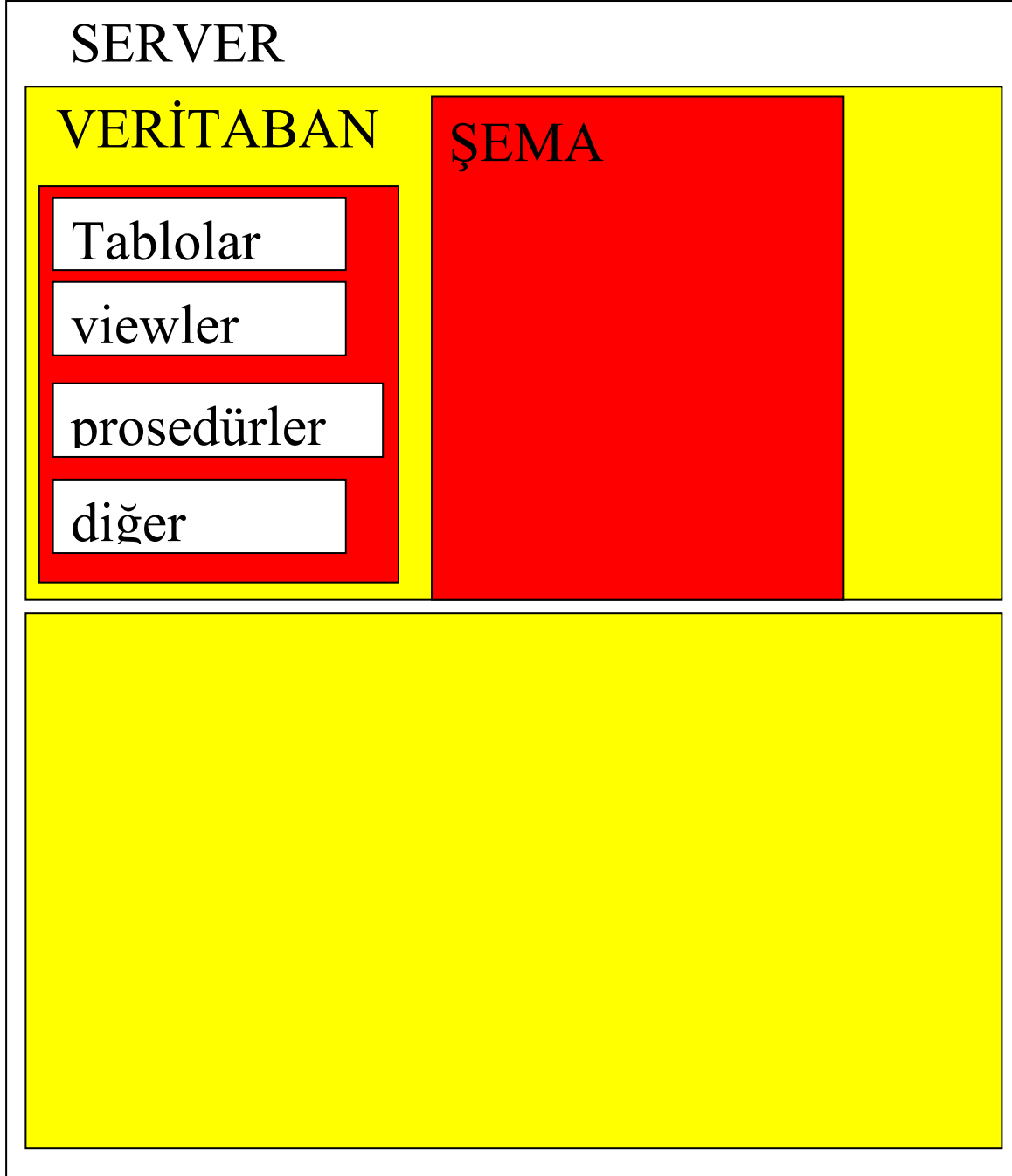
The screenshot displays a SQL Server Enterprise Manager window with two query tabs. The active tab, 'B2-30-HOCA\SQ...QLQuery8.sql\*', contains the following SQL query:

```
SELECT *,
dbo.fnDersOrtalama (DersKodu,1) as Sinav1,
dbo.fnDersOrtalama (DersKodu,2) as Sinav2
FROM Dersler
```

Below the query editor, the 'Results' pane shows a table with the following data:

	DersKodu	DersAdi	Kredisi	Sinav1	Sinav2
1	A	A Dersi	2	26,666666	60
2	B	B Dersi	4	51,75	64
3	C	C Dersi	3	50	78,666666
4	D	D Dersi	2	11	50
5	E	E Dersi	3	37,5	80
6	F	F Dersi	4	80	55

## Veritabanı yapısı ve Güvenlik



Varsayılan şemanın adı : dbo  
database owner



<http://uygulama.kilicaslan.nom.tr>  
vtys2\_h11.....sql

Tablo, view izinleri

ALTER : yapı deđiřtirme izni

CONTROL : tüm izinleri kapsıyor.

Delete : kayıt silme izni

Insert : kayıt ekleme izni

References: iliřki kurma izni

Select : listeleme izni

Update : kayıt deđiřtirme izni

Take ownership : sahiplik

Farklı řemadaki nesnelere eriřim

řemaAdi.nesneAdi

Kenan.musteri

Farklı veritabanındaki nesneye eriřim

vtAdi.şemaAdi.NesneAdi

OrnekLtd.dbo.musteri

```
create table kenan.musteri (id int, ad  
varchar(20))
```

```
insert kenan.musteri(id,ad)values  
(1,'kenan')
```

```
select * from kenan.musteri
```

```
select * from OrnekOgrenciLtd.dbo.Notlar
```