

## Örnek:

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

## Çözüm:

The image shows a screenshot of SQL Server Enterprise Manager. The left pane displays the database schema for 'OrnekOgrenciLtd'. The 'Tables' folder is expanded, showing the following tables: System Tables, dbo.DersAlanlar, dbo.Dersler, dbo.HarfliNotlar, dbo.Notlar, and dbo.Ogrenci. The right pane shows two query windows. The top window contains the query 'SELECT \* FROM Notlar' and displays the following results:

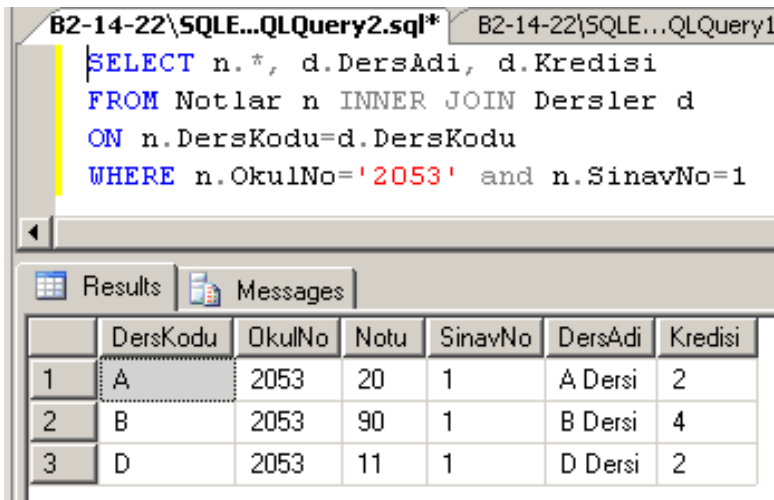
	DersKodu	OkulNo	Notu	SinavNo
1	A	2053	20	1
2	B	2053	90	1
3	D	2053	11	1

The bottom window contains the query 'SELECT \* FROM Dersler' and displays the following results:

	DersKodu	DersAdi	Kredisi
1	A	A Dersi	2
2	B	B Dersi	4
3	C	C Dersi	3
4	D	D Dersi	2
5	E	E Dersi	3
6	F	F Dersi	4

Fonksiyon yazmadan 2053 nolu öğrencinin 1 nolu sınavda aldığı notları listeleyelim.

```
SELECT n.*, d.DersAdi, d.Kredisi 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 window with the following text:

```
B2-14-22\SQL...QLQuery2.sql* B2-14-22\SQL...QLQuery1.
SELECT n.*, d.DersAdi, d.Kredisi
FROM Notlar n INNER JOIN Dersler d
ON n.DersKodu=d.DersKodu
WHERE n.OkulNo='2053' and n.SinavNo=1
```

Below the query window, the 'Results' tab is active, displaying a table with the following data:

	DersKodu	OkulNo	Notu	SinavNo	DersAdi	Kredisi
1	A	2053	20	1	A Dersi	2
2	B	2053	90	1	B Dersi	4
3	D	2053	11	1	D Dersi	2

Bu sorguyu fonksiyon içine yerleştirelim.  
CREATE FUNCTION fnOgrenciSinavNot

(@OkulNo char(10), @SinavNo tinyint)

RETURNS TABLE

AS

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

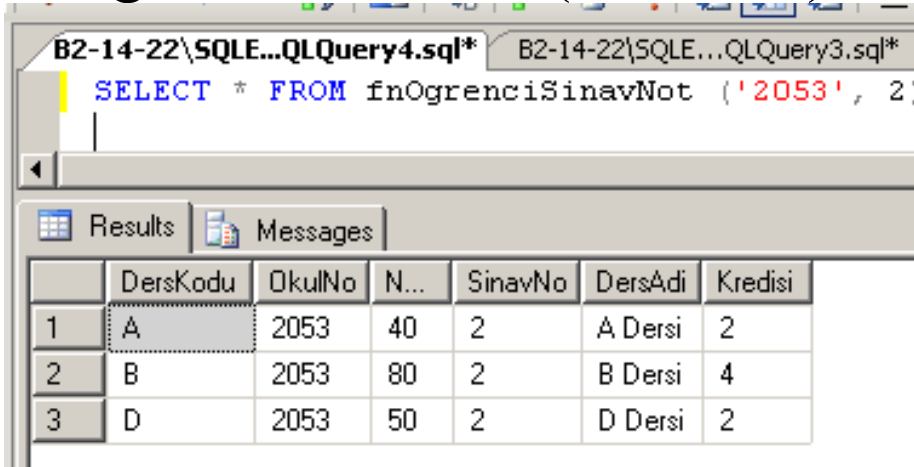
Go

Örnek çıktı

2053 nolu öğrencinin 2 nolu sınavdan aldığı notlar

SELECT \* FROM

fnOgrenciSinavNot('2053',2)



The screenshot shows a SQL Server Enterprise Manager window with two tabs: 'B2-14-22\SQL...QLQuery4.sql\*' and 'B2-14-22\SQL...QLQuery3.sql\*'. The active tab contains the SQL query: `SELECT * FROM fnOgrenciSinavNot ('2053', 2)`. Below the query window, there are two tabs: 'Results' and 'Messages'. The 'Results' tab is active and displays a table with the following data:

	DersKodu	OkulNo	N...	SinavNo	DersAdi	Kredisi
1	A	2053	40	2	A Dersi	2
2	B	2053	80	2	B Dersi	4
3	D	2053	50	2	D Dersi	2

Örnek:

Herhangi bir dersin sınav ortalamasını veren fonksiyonu oluşturunuz.

Çözüm:

Ortalamayı veren sorguyu yazalım.

A kodlu dersin 1 nolu sınavının ortalaması:

SELECT avg(Notu\*1.0) FROM Notlar  
WHERE DersKodu='A' and SinavNo=1

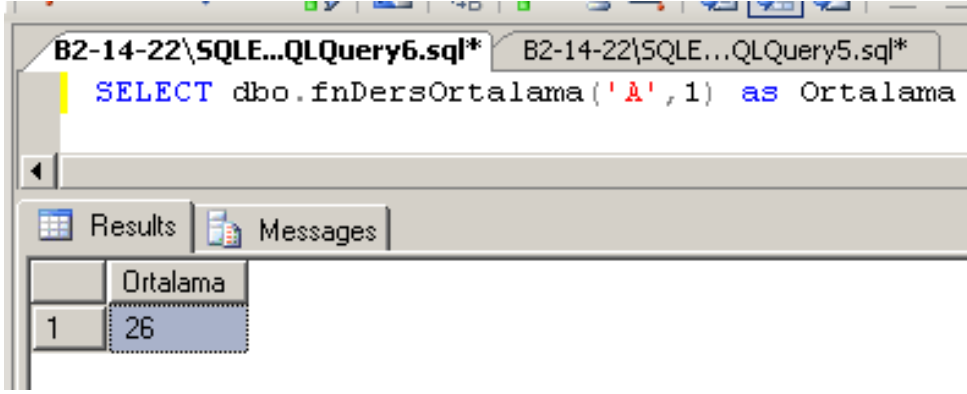
$$(1+2)/2 = 1; \quad (1.0+2.0)/2 = 1.5$$

Yukarıdaki sorguyu fonksiyon içine koyalım.

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

A kodlu dersin 1 nolu sınavının ortalaması

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



Örnek:

Yukarıdaki fonksiyonu kullanarak, tüm dersleri sınav ortalamaları ile listeleyiniz.

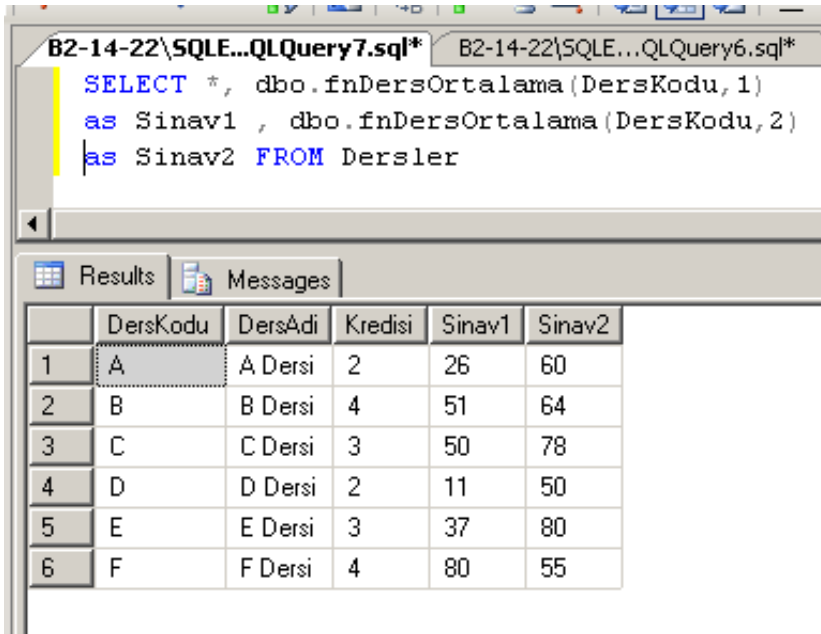
Çözüm:

SELECT \*,

**dbo.fnDersOrtalama(DersKodu,1) as**

**Sinav1 , dbo.fnDersOrtalama(DersKodu,2)**

**as Sinav2 FROM Dersler**



Örnek:

DersKodu verilen dersin harfli notlarını hesaplayan bir prosedüre yazınız.

0-40 arası FF

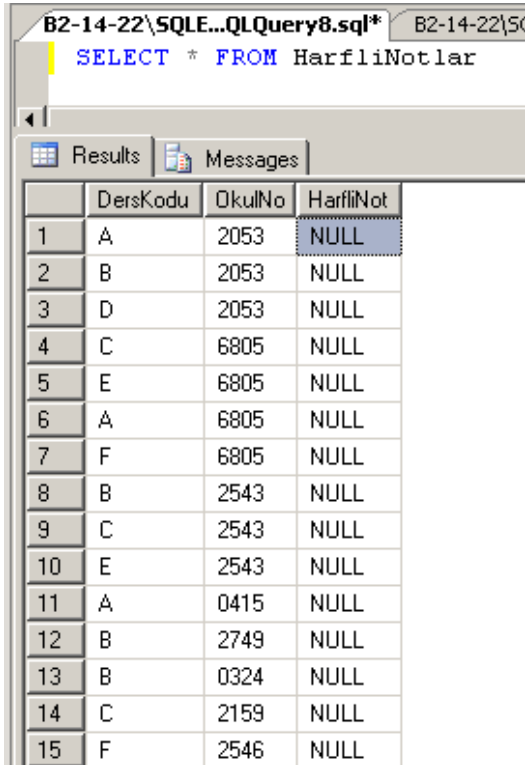
40-60 arası DD

60-70 arası CC

70-80 arası BB

80-100 arası AA

Bu prosedür her öğrencinin sınavlardan aldığı ortalamayı hesaplayacak ve yukarıdaki tabloya uygun harfi notu HarfliNotlar tablosuna işleyecek.



The screenshot shows a SQL query window with the following SQL statement: `SELECT * FROM HarfliNotlar`. Below the query window, a results grid is displayed with the following data:

	DersKodu	OkulNo	HarfliNot
1	A	2053	NULL
2	B	2053	NULL
3	D	2053	NULL
4	C	6805	NULL
5	E	6805	NULL
6	A	6805	NULL
7	F	6805	NULL
8	B	2543	NULL
9	C	2543	NULL
10	E	2543	NULL
11	A	0415	NULL
12	B	2749	NULL
13	B	0324	NULL
14	C	2159	NULL
15	F	2546	NULL

**CREATE PROCEDURE**

**spHarfliNotHesapla @DersKodu char(10)**

AS

-- @DersKodu dersini alan öğrencilere tek tek erişmeliyiz.

-- Bunun için Kürsör tanımlayacağız.

-- Kürsörden okuduğumuz okulnumarasını atamak için @OkulNo değişkeni tanımlayalım

DECLARE @OkulNo char(10)

-- Kürsörü tanımla

DECLARE notkursor CURSOR FOR

SELECT OkulNo FROM HarfliNotlar

WHERE DersKodu=@DersKodu

OPEN notkursor

FETCH NEXT FROM notkursor INTO

@OkulNo

DECLARE @ort float

WHILE @@fetch\_status=0

BEGIN

    SELECT @ort=avg(Notu) FROM  
Notlar WHERE OkulNo=@OkulNo and  
DersKodu=@DersKodu

```
UPDATE HarfliNotlar SET HarfliNot
= CASE
  WHEN @ort between 0 and 40 THEN 'FF'
  WHEN @ort between 40 and 60 THEN 'DD'
  WHEN @ort between 60 and 70 THEN 'CC'
  WHEN @ort between 70 and 80 THEN 'BB'
  WHEN @ort between 80 and 100 THEN 'AA'
END
WHERE OkulNo=@OkulNo and
DersKodu=@DersKodu
  FETCH NEXT FROM notkursor INTO
@OkulNo
END
CLOSE notkursor
DEALLOCATE notkursor
go
```

Yukarıdaki prosedürü kullanarak A kodlu dersin harfli notunu hesaplayınız.

```
EXEC spHarfliNotHesapla
@DersKodu='A'
```

B kodlu dersin harfli notunu hesaplayınız.



EXEC spHarfliNotHesapla  
@DersKodu='B'

HarfliNotlar tablosunu listeleyerek kontrol ediniz.

```
SELECT * FROM HarfliNotlar
```