

Oracle (Devam)

SELECT ... INTO ... deyimi

Sorgudan elde edilen değerlerin değişkenlere aktarılmasını sağlar. Sorgudan tek satır dönmesi gerekir, aksi durumda hata olur.

Genel ifade;

```
SELECT koloİsimleri INTO  
değişkenİsimleri FROM tabloAdi WHERE  
koşul
```

Örnek:

2053 nolu personelin adını ve soyadını değişkenlere aktarınız ve bu değişkenlerin değerini yazdırınız.

```
declare
```

```
  p_ad varchar2(20);
```

```
  p_soyad varchar2(20);
```

```
begin
```

```
select ad,soyad into p_ad,p_soyad from
personel where sicilno='2053';
dbms_output.put_line(p_ad||' '||p_soyad);
end;
```

kodu açıklayalım:

declare deyimi: değişken tanım bloğudur.

begin... end arasına: çalışacak kodlar yazılır.

p_ad,p_soyad : tanımlanan değişkenlerdir.

```
select ad,soyad into p_ad,p_soyad ....
```

Kodunda ad kolonunun değeri p_ad değişkenine, soyad kolonunun değeri p_soyad değişkenine atanır.

dbms_output: çıktıyı sağlayan oracle paketinin adı. Bu paket içindeki Put_line prosedürü ile bir satırlık çıktı yazılır.

PL/SQL de atama işlemleri

Pl/Sql de atama := sembölü ile yapılır.

isim := ad||' '||soyad;

Örnek:

2053 nolu personelin adını ve soyadını isim değişkenine aktarınız ve bu değişkeni yazdırınız.

```
declare
  p_ad varchar2(20);
  p_soyad varchar2(20);
  p_isim varchar2(30);
begin
  select ad,soyad into p_ad,p_soyad from
  personel where sicilno='2053';
  p_isim := P_ad ||' '||p_soyad;
  dbms_output.put_line(p_isim);
end;
```

atama değişken tanımlanırken de yapılabilir.

```
declare
```

Para number := 2500;

gibi

PL/SQL Blokları

- Normal blok
- Fonksiyon bloğu
- Prosedür bloğu

Normal Blok

Az önce yaptığımız uygulamalar buna örnektir.

Yapısı

declare

 tanımlamalar

begin

 komutlar

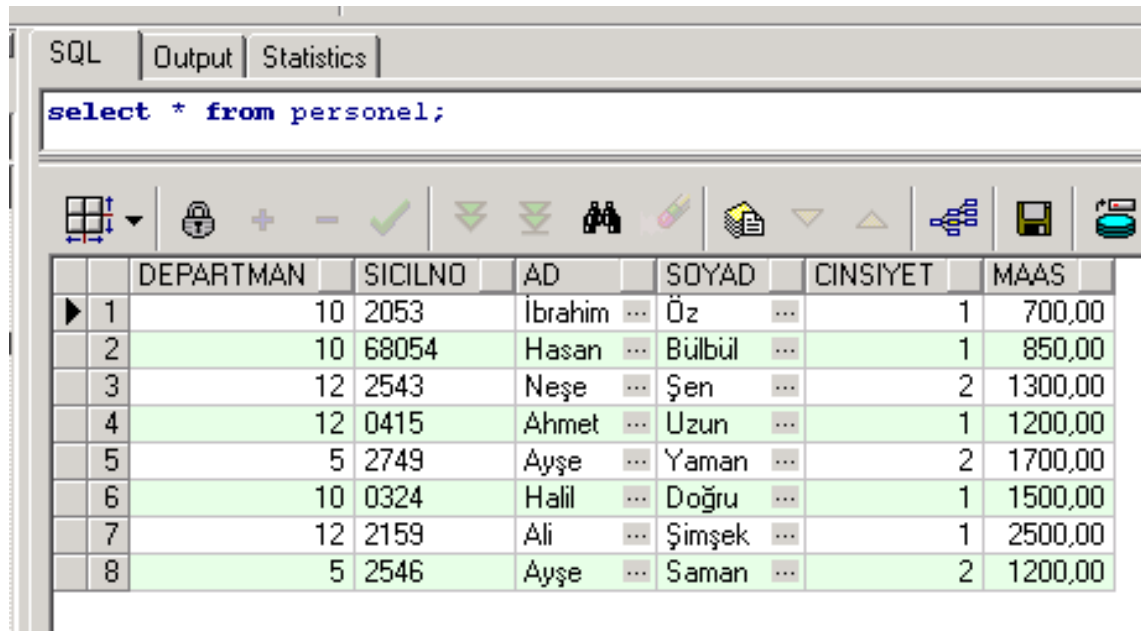
EXCEPTION

 Hata durumunda çalışacak kodlar

end;

örnek:

10 nolu departmanda çalışan personele %30 zam yapan kodu yazınız. Zam_oranı değişkeni tanımlayınız. Bu değişkene zam oranını atayınız.



The screenshot shows an Oracle SQL Developer window with the following content:

```
SQL | Output | Statistics  
select * from personel;
```

	DEPARTMAN	SICILNO	AD	SOYAD	CINSIYET	MAAS
▶ 1	10	2053	İbrahim ...	Öz ...	1	700,00
2	10	68054	Hasan ...	Bülbül ...	1	850,00
3	12	2543	Neşe ...	Şen ...	2	1300,00
4	12	0415	Ahmet ...	Uzun ...	1	1200,00
5	5	2749	Ayşe ...	Yaman ...	2	1700,00
6	10	0324	Halil ...	Doğru ...	1	1500,00
7	12	2159	Ali ...	Şimşek ...	1	2500,00
8	5	2546	Ayşe ...	Saman ...	2	1200,00

declare

 zam_orani number := 0.30;

begin

 update personel set

 maas=maas*(1+zam_orani) where
 departman=10;

 commit;

end;

fonksiyon bloğu

Oracle üzerinde fonksiyon oluşturur.
Sadece skaler değer döndürür. Tablo döndürmez.

Parametrelerde kullanılan veri tiplerine uzunluk yazılmaz. Örneğin, varchar2(20) şeklinde yazılmaz, varchar2 olarak yazılır.

create or replace function fonksiyonAdi
(parametreler)

return dönüşTipi

IS

 Değişken tanımları

begin

 komutlar

 return dönüşDeğeri

EXCEPTION

 Hata durumunda kodlar

end;

Örnek:

Sicil numarası verilen personelin maaşını veren fonksiyon oluşturunuz ve bu fonksiyon ile 2053 nolu personelin maaşını bulunuz.

```
create or replace function
maas_bul(p_sicilno char)
return number
is
  p_maas number;
begin
  select maas into p_maas from personel
where sicilno=p_sicilno;
  return p_maas;
end;
```

bu fonksiyondan yararlanarak 2053 nolu personelin maaşını yazdıralım.

```
declare
  maas number;
begin
  maas := maas_bul('2053');
```

```
dbms_output.put_line(maas);  
end;
```

prosedür bloğu

Oracle üzerinde stored prosedür oluşturur.

Yapısı

```
create or replace procedure prosedürAdi  
(parametreler)
```

```
IS
```

```
    Değişken tanımları
```

```
begin
```

```
    komutlar
```

```
exception
```

```
    hata durumunda çalışacak kodlar
```

```
end;
```

Örnek:

Personel tablosuna veri giren bir prosedür oluşturunuz.


```
create or replace procedure
personel_veri_gir
(p_sicilno char, p_adi varchar2, p_soyadi
varchar2)
is
begin
    insert into personel (sicilno,ad,soyad)
        values(p_sicilno, p_adi, p_soyadi);
    commit;
end;
```

Çalıştıralım

```
begin
    personel_veri_gir('2222', 'ali', 'veli');
end;
```