

## ORACLE AKIŞ KONTROL DEYİMLERİ

### IF ... THEN ... ELSE ... Deyimi

#### Genel ifadeler

```
IF koşul THEN  
    Komutlar  
END IF;
```

```
IF koşul THEN  
    Komutlar  
ELSE  
    Komutlar  
END IF;
```

```
IF koşul THEN  
    Komutlar  
ELSIF koşul THEN  
    Komutlar  
ELSIF koşul THEN  
    Komutlar  
ELSE  
    Komutlar  
END IF;
```

Örnek:

```
declare
```

```
  d_departman number(3);
```

```
  d_sicilno char(10):='2053';
```

```
begin
```

```
  select departman into d_departman from
  personel where sicilno=d_sicilno;
```

```
  if d_departman=5 then
```

```
    dbms_output.put_line('5 nolu departman');
```

```
  else
```

```
    dbms_output.put_line('diğer departman');
```

```
  end if;
```

```
end;
```

elsif örneği:

```
declare
```

```
  d_departman  number(3);
```

```
  d_sicilno    char(10):='2053';
```

```
begin
```

```
  select departman into d_departman from
  personel where sicilno=d_sicilno;
```

```
  if d_departman=5 then
```

```
    dbms_output.put_line('5 nolu departman');
```

```
elsif d_deptman=10 then
    dbms_output.put_line('10 nolu departman');
else
    dbms_output.put_line('diğer departman');
end if;
end;
```

## FOR DÖNGÜSÜ

### Yapısı

```
FOR kontrolDeğişkeni IN deger1..deger2
LOOP
    Komutlar
END LOOP;
```

Yukarıdaki kod kontrol değişkenini deger1 den deger2 ye kadar değiştirir. Birer birer arttırır.

Örnek:

For döngüsü ise 1 den 5'e kadar sayıları sırası ile yazdıralım.

```
declare
    i number;
begin
    for i in 1..5
```

```
loop
  dbms_output.put_line(i);
end loop;
end;
```

döngünün geriye doğru çalışması

```
FOR kontrolDeğişkeni IN REVERSE deger1..deger2
LOOP
  KOMUTLAR
END LOOP;
```

Yukarıdaki döngü deger2 den deger1'e kadar geriye doğru sayar.

```
declare
  i number;
begin
  for i IN REVERSE 1..5
  loop
    dbms_output.put_line(i);
  end loop;
end;
```

## GOTO DEYİMİ

GOTO deyimi, akışın program içinde tanımlanan bir etikete gitmesini sağlar.(LABEL) etiket <<etiketAdi>> şeklinde tanımlanır.

Yapısı

Begin

.....

<<etiketAdi>>

.....

GOTO etiketAdi;

.....

end;

Örnek:

declare

  i number;

begin

  for i in 1..10

  loop

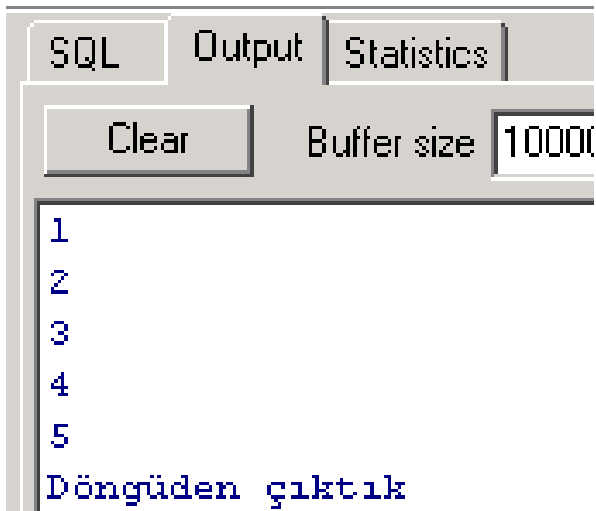
    dbms\_output.put\_line(i);

  if i=5 then

    goto a;

  end if;

```
end loop;  
dbms_output.put_line('Etiketten önceki satır');  
  
<<a>> -- a isimli etiket  
dbms_output.put_line('Döngüden çıktık');  
end;
```



## LOOP Döngüsü

Döngüden çıkış EXIT yada GOTO deyimleri ile sağlanabilir.

## LOOP

Komutlar

END LOOP;

Örnek :

1 den 5'e kadar sayıyı loop döngüsü ile yazdıralım.

```
declare
  i number;
begin
  i:=1;
  loop
    dbms_output.put_line(i);
    if i=5 then
      exit;
    end if;
    i:=i+1;
  end loop;
end;
```

## WHILE Döngüsü

Koşul sağlandığı sürece döngü devam eder. Koşul yanlış olunca döngüden çıkılır.

WHILE koşul

LOOP

Komutlar

END LOOP;

Örnek: while döngüsü ile 1 den 5'e kadar yazdıralım.

```
declare
  i number;
begin
  i:=1;

  while i<=5
  loop
    dbms_output.put_line(i);
    i:=i+1;
  end loop;
end;
```

## KAYIT VERİTİPİ:

PL/SQL de normal veritiplerinin haricinde, bir tablonun ya da bir viewin bir kaydının tutulabileceği bir veritipidir. Bı veritipinde tablonun ya da viewin bir satırı bulunur. Tablonun kolonları bu veri tipinin de kolonlarıdır.

## Tanımlama

```
değişkenAdi   tabloAdi % ROWTYPE;
```

```
p_per   personel%ROWTYPE;
```



yukarıdaki satırda p\_per personel tablosunun bir satırıdır. Buna personel tablosunun herhangi bir kaydı atanabilir.

Örnek:

2053 nolu personelin tüm bilgilerini p\_per isimli değişkene aktarınız. Bu değişkenden yararlanarak, personel adı ve soyadını yazdırınız.

```
declare
    p_per personel%rowtype;
begin
    select * into p_per from personel where
    sicilno='2053';

    dbms_output.put_line(p_per.ad||' '||
    p_per.soyad);
end;
```

*--- sql serverda aynı işi yapsaydık.*

```
declare @ad varchar(20), @soyad varchar(20)
select @ad=ad, @soyad=soyad from personel
where sicilno='2053'
print @ad+' '+@soyad
```

