
--Tüm Kolonları Seçmek

select * **from** mudurluk;

select * **from** personel;

--İstenilen Kolonları Seçmek

SELECT mudurluk_kodu, mudurluk_adi
FROM mudurluk;

--Kolonlara Alias Vermek

SELECT ad, soyad,
 maas "Maaş",
 (maas*5/100) vergi,
 semt **as** oturugu_semt
FROM personel;

--Aritmetik Operatörlerin Kullanımı

SELECT ad, soyad, maas,
 (maas + maas * 8 / 100) zamli_maas
FROM personel;

select
 ad, soyad,
 giris_tarihi + 7 **as** oryantasyon_gunu,
 30 - izin_gunu **as** kalan_izin_gun_sayisi
from personel;

--NULL Değeri ve Özellikleri

SELECT ad, soyad, semt, maas, prim,
 (maas + prim) hesaba_yatan
FROM personel

--Birleştirme Operatörü

```
SELECT ad, soyad,  
       ad||' '||soyad adi_soyadi,  
       concat(ad,soyad) adsoyad,  
       zmaas || ' TL' maasi  
FROM personel;
```

```
SELECT ad, soyad, prim, semt,  
       'Prim: [' || prim || ']' Semt: [' || semt || ']' prim_semt  
FROM personel
```

--Tekrarlayan Satırlar - Distinct

```
SELECT DISTINCT unvan, semt  
FROM personel
```

--Tablo Yapısını Görüntüleme

```
DESCRIBE personel;
```

```
DESC personel;
```

--ALIŞTIRMALARIN CEVAPLARI

--Birleştirme Operatörü

```
select ad, soyad,  
       ad || '.' || soyad || '@dbhunter.net' as email,  
       maas as eski_maas, maas*1.45 as yeni_maas  
from personel;
```

--Tekrarlayan Satırlar - Distinct

```
select distinct personel_id  
from personel_izinleri;
```

--WHERE ifadesinin Kullanımı

```
SELECT * FROM mudurluk
WHERE il_adi = 'ANKARA'
```

--Karşılaştırma Operatörleri

```
SELECT ad, soyad, maas FROM personel
WHERE maas >= 3200
```

--BETWEEN Operatörü

```
SELECT ad, soyad, maas FROM personel
WHERE maas between 1000 and 2000
```

--IN Operatörü

```
SELECT personel_id, ad, soyad, maas
FROM personel
WHERE personel_id in (5000, 5008, 5013)
```

--LIKE Operatörü

```
select ad, soyad, maas
from personel
where ad like 'A%'
```

```
select ad, soyad, maas
from personel
where ad like '_K%'
```

```
select ad, soyad, maas
from personel
where ad like '%U_'
```

```
select ad, soyad, maas
from personel
where soyad like '%M%'
```

--IS NULL Operatörü

```
select ad, soyad, semt from personel
where semt is null
```

--AND Operatörü

```
select ad, soyad, maas, unvan
from personel
where unvan = 'MÜHENDİS'
      and maas > 2500
```

--OR Operatörü

```
select ad, soyad, maas, unvan
from personel
where unvan = 'MÜHENDİS'
      or maas > 2500
```

--NOT Operatörü

```
select ad, soyad, maas, unvan
from personel
where unvan not in ('MÜHENDİS', 'UZMAN')
```

```
select ad, soyad, maas, unvan
from personel
where ad not like 'A%'
```

--Operatör Üstünlük Sırası

```
select ad, soyad, maas, unvan
from personel
where (unvan = 'MÜHENDİS'
      or
      unvan = 'UZMAN')
      and
      maas > 2000
```

--Order By Operatörü

```
select ad, soyad, maas, unvan
from personel
order by maas
```

```
select *
from personel
order by 5
```

```
select ad, soyad, maas, unvan
from personel
order by unvan, maas desc
```

```
select *
from personel
order by konum_id nulls last
```

```
select *
from personel
order by konum_id nulls first
```

--Dışardan Değişken Atama

```
select personel_id, ad, soyad, maas, unvan
from personel
where personel_id = &id
```

```
select personel_id, ad, soyad,
      maas, &kolon_ismi
from personel
where &sart
order by &sira_sutun
```

--ALIŞTIRMALARIN CEVAPLARI

--BETWEEN Operatörü

--Alıştırma-1

```
select * from personel  
where ad between 'B' and 'E'
```

--Alıştırma-2

```
select * from personel  
where '01.01.2010' between giriş_tarihi and çıkış_tarihi
```

--IN Operatörü

```
select * from personel  
where semt in ('ACIBADEM', 'ÜMRANIYE')
```

--LIKE Operatörü

--Alıştırma-1

```
select * from personel  
where ad like '%A%T%'
```

--Alıştırma-2

```
select * from personel  
where ad like '_AH%'
```

--AND Operatörü

```
select * from personel  
where dept_id = 113  
      and maaş > 4000  
      and semt <> 'ÇUKURAMBAR'
```

--NOT Operatörü

```
select ad, soyad, maas, dept_id
from personel
where 1=1
      and not dept_id = 102
      and not maas between 3000 and 5000
order by maas
```

--Operatör Üstünlük Sırası

```
select ad, soyad, maas, unvan
from personel
where unvan not in ('MÜHENDİS', 'UZMAN')
      and (ad like 'A%' or ad like '%A')
```

--Büyük-Küçük Karakter Dönüşüm Fonksiyonları

```
select ad,
       lower(ad) kucuk_harf,
       upper(ad) buyuk_harf,
       initcap(ad) kelime_buyuk
from personel;
```

```
select * from personel
where ad = 'serkan'
```

```
select * from personel
where lower(ad) = 'serkan'
```

--Karakter İşleme Fonksiyonları

```
select concat(ad, soyad) birlestirme,
       substr(ad, 1, 5) sinirlama,
       length(ad) uzunluk,
       instr(ad, 'A') nerede,
       lpad(ad, 12, '-') sola_doldur,
       rpad(ad, 12, '-') saga_doldur,
       replace(ad, 'E', '*') degistir
from personel
```

--Sayı Fonksiyonları

```
select maas,
       round(maas), round(maas,1), round(maas,-1),
       trunc(maas), trunc(maas,1),
       floor(maas),
       mod(maas,100)
from personel;
```

--Tarih Fonksiyonları

```
select months_between('01.07.2011', '01.01.2011') kac_ay,
       sysdate simdi,
       add_months(sysdate,5) ay_ekle,
       next_day(sysdate, 'CUMA') sonraki_cuma,
       last_day(sysdate) ayin_son_gunu,
       round(sysdate-60, 'MONTH') r1,
       round(sysdate, 'YEAR') r2,
       trunc(sysdate, 'MONTH') t1,
       trunc(sysdate, 'YEAR') t2
from dual;
```

--Tarihler ile Aritmetik İşlemler

```
select ad, soyad, giris_tarihi,  
       giris_tarihi - 10,  
       giris_tarihi + 2/24  
From personel
```

--TO_CHAR Fonksiyonu (Tarih)

```
select giris_tarihi,  
       to_char(giris_tarihi, 'dd.mm.yyyy') fm1,  
       to_char(giris_tarihi, 'DD Month YYYY') fm2,  
       to_char(giris_tarihi, 'Ddspth Month YYYY HH:MI:SS') fm3,  
       to_char(giris_tarihi, 'MM') fm4,  
       to_char(giris_tarihi, 'YYYY') fm5,  
       to_char(giris_tarihi, 'DAY') fm6,  
       to_char(giris_tarihi, 'Q') fm7  
from personel
```

```
SELECT customer_id, order_date,  
       TO_CHAR(order_date, 'HH24:MI:SS') as TC1,  
       TO_CHAR(order_date, 'MON-DD-YYYY HH12:MI PM') as TC2,  
       TO_CHAR(order_date, 'DD.MM.YYYY HH24:MI') as TC3,  
       TO_CHAR(order_date, 'MON-DAY-YYYY HH12:MI') as TC4,  
       TO_CHAR(order_date, 'Month DD, YYYY') as TC5,  
       TO_CHAR(order_date, 'YYYYMMDD') as TC6  
FROM orders;
```

--TO_CHAR Fonksiyonu (Sayı)

```
select maas,  
       to_char(maas, '999999.99') fm1,  
       to_char(maas, '999,999.99') fm2,  
       to_char(maas, '9990D000000') fm3,  
       to_char(maas, '$99,999.00') fm4  
from personel
```

--TO_NUMBER ve TO_DATE Fonksiyonları

```
select to_date('21.06.2011', 'dd.mm.yyyy') tarih,
       to_number('00001') sayi1,
       to_number('123.45', '9999.99') sayi2
from dual
```

```
SELECT TO_DATE('20230405', 'YYYYMMDD') TD1,
       TO_DATE('2023 APRIL 05', 'YYYY MONTH DD') TD2,
       TO_DATE('2023 april 05', 'YYYY month DD') TD3,
       TO_DATE('2023 APR 05', 'YYYY MON DD') TD4,
       TO_DATE('2023 240', 'YYYY DDD') TD5,
       TO_DATE('February 08, 2023', 'Month DD, YYYY') TD6
FROM DUAL;
```

```
SELECT TO_NUMBER('123,45') TN1,
       TO_NUMBER('123.45', '9999.99') TN2,
       TO_NUMBER('$12,123.23', '$999,999.99') TN3,
       TO_NUMBER('- $12,345.67', '$99,999.99') TN4,
       TO_NUMBER('00005469', '9999999999') TN5
FROM DUAL
```

--CAST Fonksiyonu

```
select
  cast('100' as integer) as cast1,
  cast('01/01/2021' as date) as cast2,
  cast('15/oct/2022' as date) as cast3,
  cast('10,25' as float) as cast4,
  cast(876498 as varchar2(10)) cast5,
  cast('22-jun-2023' as timestamp with local time zone) as cast6,
  cast('5 10:5:4' as interval day to second) as cast7
from dual;
```

--İç-İçe Fonksiyonlar

```
select semt,
       substr(semt, instr(lower(semt), 't'), 2) deger
from personel
```

--NVL Fonksiyonu

```
select semt,
       nvl(semt, 'Semt Bilgisi Girilmemiş') deger
from personel
```

```
-----  
--NULLIF Fonksiyonu  
-----
```

```
select mudurluk_adi, musteri_sayisi, musteri_hedefi,  
       nullif(musteri_sayisi, musteri_hedefi) hedef_olmusmu  
from mudurluk_hedef
```

```
-----  
--COALESCE Fonksiyonu  
-----
```

```
select coalesce(null, 1, null, 4) deger from dual;  
  
select personel_id, giris_tarihi, cikis_tarihi,  
       coalesce(giris_tarihi, cikis_tarihi) deger  
from personel;
```

```
-----  
--DECODE Fonksiyonu  
-----
```

```
select ad, soyad, unvan, maas,  
       decode  
       (  
         unvan,  
         'MÜHENDİS', maas*0.05,  
         'UZMAN',    maas*0.02,  
         maas*0.03  
       ) kesinti  
from personel
```

```
-----  
--CASE İfadesi  
-----
```

```
select maas,  
       case  
         when maas < 2000 then 'Düşük Maaş'  
         when maas >= 2000 and maas <= 5000 then 'Orta Maaş'  
         when maas > 5000 then 'Yüksek Maaş'  
       end seviye  
from personel
```

--ALIŞTIRMALARIN CEVAPLARI

--Büyük-Küçük Karakter Dönüşüm Fonksiyonları

```
select dept_id, upper(dept_ismi)
from departman
where lower(dept_ismi) like '%uygulama%'
```

--Karakter İşleme Fonksiyonları

```
select * from orders
where ship_address like '%Ave%';

select * from orders
where initcap(ship_address) like '%Ave%';

select * from orders
where instr(ship_address, 'Ave') > 0;
```

--Sayı Fonksiyonları

```
select p.unit_price,
       round(unit_price,1) as unit_price_d1,
       round(units_in_stock/20) as package_count,
       mod(units_in_stock,20) as remaining_amount
from products p;
```

--Tarih Fonksiyonları

```
select giris_tarihi,
       add_months(giris_tarihi, 180) as tazminat_hakedis_tarihi,
       maas * 1.5 as tazminat_miktari
from personel
where giris_tarihi is not null
     and cikis_tarihi is null;
```

--TO_CHAR Fonksiyonu

```
select
'Müşteri: ' || customer_id ||
', Sipariş Tarihi: ' || to_char(order_date, 'mm-dd-yyyy') ||
', Gönderilme Tarihi: ' || to_char(shipped_date, 'mm-dd-yyyy') ||
', Kargo Süresi (gün): ' || to_char(shipped_date - order_date) ||
', Ücret: ' || to_char(freight, '9990D00') as siparis_bilgi
from orders ;
```

--COALESCE Fonksiyonu

```
select
    ad, soyad, maas, prim,
    maas + coalesce(prim,0) hesaba_yatan
from personel;
```

--DECODE Fonksiyonu

```
select order_id, customer_id,
    decode(ship_via, 1, 'Hava Yolu', 2, 'Deniz Yolu', 3, 'Kara Yolu') as ship_type
from orders;
```

--DECODE vs CASE

--Soru

```
select semt,
    decode(semt,
        'BATIKENT', 'YM',
        'DEMET', 'YM',
        'ÜMİTKÖY', 'YM',
        'ÖVEÇLER', 'ÇN',
        'ÇUKURAMBAR', 'ÇN',
        'DİKMEN', 'ÇN',
        'ANKARA') ilce
from personel;
```

--Cevap

```
select semt,
    case
        when semt in ('BATIKENT', 'DEMET', 'ÜMİTKÖY') then 'YM'
        when semt in ('ÖVEÇLER', 'ÇUKURAMBAR', 'DİKMEN') then 'ÇN'
        else 'ANKARA'
    end ilce
from personel;
```

--CASE İfadesi

```
select company_name, contact_name, contact_title,
    case
        when lower(contact_title) like '%manager%' then 'Manageral Position'
        when lower(contact_title) like '%admin%' then 'Administrative Position'
        when lower(contact_title) like '%asistant%' then 'Assistant Position'
        when lower(contact_title) like '%representative%' then 'Representative Position'
        else 'Other Position'
    end as position_name
from customers;
```

--Tek Satır Fonksiyonları - Mix

```
select ad, soyad,  
       to_char(giris_tarihi, 'dd month yyyy') giris,  
       decode(prim, null, 'Normal', 'Torpilli') seviye,  
       initcap(unvan) unvan  
from personel  
where instr(lower(unvan), 'd') > 0
```

--AVG ve SUM

```
select round(avg(maas),2) maas_ortalaması,  
       sum(maas) toplam_maas  
from personel;
```

```
select  
    sum(freight), sum(distinct freight),  
    avg(freight), avg(distinct freight)  
from orders
```

--MIN ve MAX

```
select min(maas) en_dusuk_maas,  
       max(maas) en_yuksek_maas,  
       min(giris_tarihi) ilk_ise_baslama,  
       max(giris_tarihi) son_ise_baslama  
from personel  
where unvan = 'MÜHENDİS'
```

--COUNT

```
select count(*) kayıt_sayisi,  
       count(prim) prim_alani_null_olmayan  
from personel  
where unvan = 'UZMAN'
```

--DISTINCT

```
select distinct unvan  
from personel  
order by unvan;
```

```
select count(*) tum_kayıt_sayisi,  
       count(semi) semi_null_olmayan,  
       count(distinct semi) kac_farkli_semi  
from personel;
```

--GROUP BY İfadesi

```
select semt
from personel
group by semt
order by 1
```

--GROUP BY – Bir Kolon

```
select unvan,
       count(*) personel_sayisi,
       round(avg(maas),2) maas_ortalamasi,
       sum(maas) toplam_maas
from personel
group by unvan
order by unvan;
```

--GROUP BY – Birden Fazla Kolon

```
select unvan, semt,
       count(*) personel_sayisi
from personel
group by unvan, semt
order by unvan, semt;
```

--HAVING İfadesi

```
select unvan ,
       round(avg(maas),2) maas_ortalamasi
from personel
group by unvan
having avg(maas) > 3000
order by unvan;
```

--ALIŞTIRMALARIN CEVAPLARI

--AVG ve SUM

```
SELECT
    SUM(shipped_date - order_date) as toplam_gun,
    AVG(shipped_date - order_date) as ortalama_gun
FROM orders
where ship_city = 'Stuttgart'
```

--MIN - MAX - AVG

```
select min(length(concat(ad,soyad))) enaz,
       max(length(concat(ad,soyad))) encok,
       round(avg(length(concat(ad,soyad))),2) ortalama
from personel;
```

--COUNT

```
select
    count(*) as kayit_sayisi,
    count(distinct discount) as farkli_indirim_orani_sayisi
from order_details od
```

--GROUP BY İfadesi

--Alıştırma-1

```
select customer_id,
       count(*) as siparis_adi,
       count(distinct ship_city) as farkli_sehir_sayisi,
       min(order_date) as ilk_siparis_tarihi,
       max(order_date) as son_siparis_tarihi
from orders
group by customer_id;
```

--Alıştırma-2

```
select
    sum(case ship_via when 1 then 1 else 0 end) as hava_yolu_adi,
    sum(case ship_via when 2 then 1 else 0 end) as deniz_yolu_adi,
    sum(case ship_via when 3 then 1 else 0 end) as kara_yolu_adi
from orders
```

```
--HAVING
```

```
--Alıştırma-1
```

```
select semt, count(*)
from personel
group by semt
having count(*) > 5
order by semt;
```

```
--Alıştırma-2
```

```
select supplier_id, count(*)
from products
group by supplier_id
having count(*) > 3;
```

```
--Grup Fonksiyonları - Mix
```

```
select konum_id,
       count(*) personel_sayisi,
       sum(maas) toplam_maas,
       round(avg(maas),2) ortalama_maas
from personel
group by konum_id
having count(*) > 10
```

--Inner Join - (Eşiti Olan Birleştirme)

```
select pr.ad, pr.soyad, pr.konum_id, kn.konum_adi
from personel pr, konum kn
where pr.konum_id = kn.konum_id
```

```
select pr.ad, pr.soyad, pr.konum_id, kn.konum_adi
from personel pr inner join konum kn
    on pr.konum_id = kn.konum_id
```

```
select md.mudurluk_kodu, md.mudurluk_adi, sb.sube_adi
from mudurluk md, mudurluk_sube sb
where md.mudurluk_kodu = sb.mudurluk_kodu;
```

```
select md.mudurluk_kodu, md.mudurluk_adi, sb.sube_adi
from mudurluk md
inner join mudurluk_sube sb
    on md.mudurluk_kodu = sb.mudurluk_kodu;
```

--Outer Join - (Dış Birleştirme)

```
select ad, soyad, konum_adi, kn.konum_id
from personel pr, konum kn
where pr.konum_id = kn.konum_id(+)
order by konum_adi nulls first
```

```
select ad, soyad, konum_adi, kn.konum_id
from personel pr left outer join konum kn
    on pr.konum_id = kn.konum_id
order by konum_adi nulls first
```

--Right Join

```
SELECT
    id_a, fruit_a,
    id_b, fruit_b
FROM basket_a
RIGHT JOIN basket_b
    ON fruit_a = fruit_b;
```

--Full Join

```
SELECT
    id_a, fruit_a,
    id_b, fruit_b
FROM basket_a
FULL JOIN basket_b
    ON fruit_a = fruit_b;
```

--Eşiti Olmayan Birleştirme

```
select pr.ad, pr.soyad, pr.maas, uc.derece, uc.aciklama
from personel pr, ucret_duzey uc
where pr.maas between uc.maas_alt_limit and uc.maas_ust_limit
order by 1,2
```

--Self Join - (Kendine (İç) Birleştirme)

```
select pr.ad, pr.soyad,
    pry.ad || ' ' || pry.soyad yönetici
from personel pr, yönetici yn, personel pry
where pr.yoneticisi_id = yn.yoneticisi_id
    and pry.personel_id = yn.personel_id
order by 3;
```

```
select
    e.first_name as emp_fname, e.last_name as emp_lname,
    r.first_name as rep_fname, r.last_name as rep_lname,
from employees e, employees r
where e.reports_to = r.employee_id;
```

--Kartezyen Çarpım

```
select ad, soyad, pr.konum_id, kn.konum_adi
from personel pr, konum kn
--where kn.konum_id = pr.konum_id
order by 1,2
```

```
SELECT brand, color
FROM cars, colors;
```

```
SELECT brand, color
FROM cars
CROSS JOIN colors;
```

--ALIŞTIRMALARIN CEVAPLARI

--Inner Join (Eşiti Olan Birleştirme)

--Alıştırma-1

```
select
    c.customer_id, c.company_name,
    o.order_date, o.shipped_date, o.ship_name
from customers c, orders o
where o.customer_id = c.customer_id;

select
    c.customer_id, c.company_name,
    o.order_date, o.shipped_date, o.ship_name
from customers c
inner join orders o on o.customer_id = c.customer_id;
```

--Alıştırma-2

```
select
    c.customer_id, c.company_name,
    o.order_date, p.product_name
from customers c, orders o,
    order_details od, products p
where c.customer_id = o.customer_id
    and o.order_id = od.order_id
    and p.product_id = od.product_id
```

--Left Join

```
select md.mudurluk_kodu, md.mudurluk_adı, sb.sube_adı
from mudurluk md, mudurluk_sube sb
where md.mudurluk_kodu = sb.mudurluk_kodu (+);

select md.mudurluk_kodu, md.mudurluk_adı, sb.sube_adı
from mudurluk md left outer join mudurluk_sube sb
    on md.mudurluk_kodu = sb.mudurluk_kodu;
```

--Kartezyen Çarpım

```
select c1.brand, c2.type_name, c3.color
from cars c1, car_types c2, colors c3
order by 1,2,3
```

```
select c1.brand, c2.type_name, c3.color
from cars c1
  cross join car_types c2
  cross join colors c3
order by 1,2,3
```

--UNION Operatörü

```
SELECT fruit_a FROM basket_a
UNION
SELECT fruit_b FROM basket_b
```

```
select ad, soyad, maas, unvan from personel
Where maas between 1200 and 2500
      and konum_id = 1
union
select ad, soyad, maas, unvan from personel
Where maas between 1800 and 5000
      and konum_id = 1
```

--UNION ALL Operatörü

```
select ad, soyad, maas, unvan from personel
Where maas between 1200 and 2500
      and konum_id = 1
union all
select ad, soyad, maas, unvan from personel
Where maas between 1800 and 5000
      and konum_id = 1
order by 1
```

```
SELECT fruit_a FROM basket_a
UNION ALL
SELECT fruit_b FROM basket_b
```

--INTERSECT Operatörü

```
select ad, soyad, maas, unvan from personel
Where maas between 1200 and 2500
      and konum_id = 1
intersect
select ad, soyad, maas, unvan from personel
Where maas between 1800 and 5000
      and konum_id = 1
```

--MINUS Operatörü

```
select ad, soyad, maas, unvan from personel
Where maas between 1800 and 5000
      and konum_id = 1
```

minus

```
select ad, soyad, maas, unvan from personel
Where maas between 1200 and 2500
      and konum_id = 1
```

```
select personel_id from personel
minus
select personel_id from ceza_bilgi
order by 1
```

```
select personel_id from personel
minus
select personel_id from personel_izinleri
order by 1
```

```
SELECT fruit_a FROM basket_a
MINUS
SELECT fruit_b FROM basket_b
```

--ALIŞTIRMALARIN CEVAPLARI

--UNION Operatörü

```
select country, company_name, count(*) as order_count
from customers c, orders o
where country = 'France'
      and c.customer_id = o.customer_id
group by country, company_name, c.customer_id
union
select country, company_name, count(*) as order_count
from customers c, orders o
where country = 'Germany'
      and c.customer_id = o.customer_id
group by country, company_name, c.customer_id
```

--UNION ALL Operatörü

```
select 'Customer' as type_name, company_name
from customers
union all
select 'Shippers' as type_name, company_name
from shippers
union all
select 'Suppliers' as type_name, company_name
from suppliers
```

--INTERSECT Operatörü

```
select s.company_name
from products p, suppliers s
where category_id = 2
      and s.supplier_id = p.supplier_id
intersect
select s.company_name
from products p, suppliers s
where category_id = 5
      and s.supplier_id = p.supplier_id
```

--MINUS Operatörü

```
select customer_id from customers c  
minus  
select customer_id from orders o
```

--Tek Satır Alt Sorgular

```
select ad, soyad, maas, unvan
from personel
where unvan =
      (select unvan from personel where personel_id = 5025)
```

```
select ad, soyad, maas, unvan
from personel
where maas >
      (select maas from personel where personel_id = 5007)
```

```
select ad, soyad, maas, unvan
from personel
where maas = (select min(maas) from personel)
```

--Tek Satır Alt Sorgular - Having

```
select
      (select konum_adi from konum kn
       where kn.konum_id=ps.konum_id) konm,
count(*) pers_sayisi
from personel
group by konum_id
having count(*) > (select count(*) from personel where konum_id = 3)
```

--IN Operatörü

```
select * from personel
where personel_id in
      (select personel_id from yonetici where seviye=1)
```

```
select customer_id, company_name, contact_name
from customers
where customer_id in (select customer_id from orders)
```

```
select * from personel
where (yonetici_id, unvan) in
(
    select yonetici_id, unvan
    from personel where personel_id = 5001
)
```

--ANY Operatörü

```
select ad, maas from personel
where maas > any (3000, 4000, 5000)
order by 2
```

```
select ad, maas from personel
where maas > 3000 or maas > 4000 or maas > 5000
```

```
select ad, soyad, unvan, maas from personel
where maas < any (select maas from personel where unvan = 'UZMAN')
```

--ALL Operatörü

```
select ad, maas from personel
where maas > all (3000, 4000, 5000)
order by 2
```

```
select ad, maas from personel
where maas > 3000 and maas > 4000 and maas > 5000
```

```
select ad, soyad, unvan, izin_gunu from personel
where izin_gunu > all
(select izin_gunu from personel where unvan = 'GRUP MÜDÜRÜ')
```

--EXISTS

```
select * from personel pr
where pr.yonetici_id in
(
    select yn.yonetici_id from yonetici yn
    where seviye = 1
)
```

```
select * from personel pr
where exists
(
    select 1 from yonetici yn
    where seviye = 1
        and yn.yonetici_id = pr.yonetici_id
)
```

--NOT EXISTS

```
select * from personel pr
where not exists
(
    select 1 from yonetici yn
    where seviye = 1
        and yn.yonetici_id = pr.yonetici_id
)
```

--WITH

```
with ort_maas as
(
    select yonetici_id, round(avg(maas),2) ortalama
    from personel
    where yonetici_id is not null
    group by yonetici_id
)
select ad, soyad, maas, pr.yonetici_id
from personel pr, ort_maas om
where pr.yonetici_id = om.yonetici_id
    and om.ortalama > 4000
```

```
with dept_costs as
(
    select department_name,
           sum(salary) dept_total
    from employees emp, departments dep
    where
        emp.department_id = dep.department_id
    group by department_name
),
avg_costs as
(
    select
        sum(dept_total) / count(*) dept_avg
    from dept_costs
)
select *
from dept_costs
where dept_total >
    ( select dept_avg from avg_costs)
order by department_name;
```

--ALIŞTIRMALARIN CEVAPLARI

--Tek Satır Alt Sorgular

```
select p.ad, p.soyad,
(
    select d.dept_ismi from departman d
    where d.dept_id = p.dept_id
) as dep_ismi
from personel p
```

--IN Operatörü

--Alıştırma-1

```
select customer_id, company_name, contact_name
from customers
where customer_id in
(
    select customer_id from orders
    group by customer_id
    having count(*) > 10
)
```

--Alıştırma-2

```
select c.customer_id, company_name, contact_name, order_count
from customers c,
(
    select customer_id, count(*) as order_count
    from orders
    group by customer_id
    having count(*) > 10
) order_info
where c.customer_id = order_info.customer_id;
```

--Alıştırma-3

```
select * from employees e
where e.employee_id in
(
    select et.employee_id from employee_territories et
    where et.territory_id in
    (
        select t.territory_id from territories t
        where t.region_id in
        (
            select r.region_id from region r
            where r.region_description = 'Eastern'
        )
    )
)
```

--ANY Operatörü

```
select * from personel
where maas < (select max(maas) from personel where unvan = 'UZMAN')

select * from personel ps1
where exists (select 1 from personel ps2 where unvan = 'UZMAN' and ps1.maas
< ps2.maas)
```

--ALL Operatörü

```
select customer_id, order_date,
       freight, ship_country
from orders
where 1=1
      and ship_country <> 'Mexico'
      and freight > all
(
    select freight from orders o
    where o.ship_country = 'Mexico'
)
```

--EXISTS Operatörü

--Alıştırma-1

```
select * from customers c
where not exists
(
    select 1 from orders o
    where o.customer_id = c.customer_id
)
```

--Alıştırma-2

```
select * from shippers s
where exists
(
    select 1 from orders o
    where s.shipper_id = o.ship_via
)
```

--Alıştırma-3

```
select customer_id, company_name, contact_name
from customers c
where exists
(
    select customer_id from orders o
    where c.customer_id = o.customer_id
    group by customer_id
    having count(*) > 10
)
```

```
-----  
--WITH  
-----
```

```
--Alıştırma-1
```

```
select ad, soyad, maas, pr.yonetici_id  
from personel pr,  
(  
  select yonetici_id, round(avg(maas),2) ortmaas  
  from personel  
  where yonetici_id is not null  
  group by yonetici_id  
) om  
where pr.yonetici_id = om.yonetici_id  
      and om.ortmaas > 4000  
  
-----
```

```
--Alıştırma-2
```

```
with order_info as  
(  
  select customer_id, count(*) as order_count  
  from orders  
  group by customer_id  
  having count(*) > 10  
)  
select c.customer_id, company_name,  
       contact_name, order_count  
from customers c, order_info o  
where c.customer_id = o.customer_id;
```

--INSERT - Bir Tabloya Kayıt Ekleme

```
insert into konum(konum_id, konum_adi, il_kodu)
values (10, 'Ankara-Yenimahalle', 6);
```

```
insert into konum
values (11, 'İstanbul-Pendik', 34);
```

```
insert into ceza_bilgi
values(5021, 'Yüz Kızartıcı Suç', NULL, NULL);
```

```
insert into ceza_bilgi
values(6761, 'Uzaklaştırma', sysdate, to_date('31.12.2011',
'DD.MM.YYYY'));
```

--INSERT - Bir Tabloya Kayıt Ekleme - Default Değer

```
create table default_ornek
(
    ad varchar2(20),
    soyad varchar2(30) default '--',
    tarih date
);
```

```
insert into default_ornek(ad, soyad, tarih)
values('Ali', default, sysdate);
```

```
select * from default_ornek;
```

```
insert into default_ornek (ad,soyad,tarih)
values(default, default, sysdate);
```

```
INSERT INTO student(id, name, class_no, mark, gender, course_name)
VALUES(36, 'Adele', DEFAULT, DEFAULT, DEFAULT, 'Finance');
```

```
INSERT INTO student(id, name, class_no, mark, gender, course_name)
VALUES(37, 'Tarkan', DEFAULT, 78, 'male', 'Zoology');
```

```
INSERT INTO student(id, name, mark, course_name)
VALUES(38, 'Ava Max', 85, 'History');
```

--UPDATE - Verileri Güncelleme

```
update personel
set maas = 2800
where personel_id = 5012
```

```
update personel
set giris_tarihi = to_date('01.01.2012', 'DD.MM.YYYY')
where yonetici_id = 919
```

```
update personel set
maas = (select maas from personel where personel_id = 5004),
unvan = (select unvan from personel where personel_id = 5004)
where personel_id = 5005
```

```
update personel set
(maas, unvan)
=
(select maas, unvan from personel where personel_id = 5004)
where personel_id = 5005
```

```
update product p
set net_price = price - price *
(
    select s.discount from product_segment s
    where p.segment_id = s.id
)
```

--DELETE - Verileri Silme

```
delete from mudurluk where mudurluk_kodu = 62
```

```
delete from mudurluk_sube
where mudurluk_kodu in
(
    select mudurluk_kodu from mudurluk
    where il_adi = 'HATAY'
)
```

```
delete from personel p
where p.dept_id in
(
    select d.dept_id from departman d
    where upper(d.dept_ismi) like '%CRM%'
)
```

```
delete from student
where id = 36;
```

```
delete from basket_c
where id_c in (8, 9);
```

```
delete from student
where name = 'arnold';
```

```
delete from basket_c
where id_c = 5;
```

--ALIŞTIRMALARIN CEVAPLARI

--INSERT – Bir Tabloya Kayıt Ekleme

```
INSERT INTO student(id, name, class, mark, gender, course_name, email)
```

```
VALUES(40, 'Zeki Müren', 2, 85, 'male', 'Conservatory', null);
```

```
INSERT INTO student
```

```
VALUES(41, 'Emel Sayın', 4, 96, DEFAULT, 'Finance', default);
```

--UPDATE – Verileri Güncelleme

--Alıştırma-1

```
update personel p
```

```
set maas = maas * 1.25
```

```
where p.dept_id in
```

```
(
```

```
    select d.dept_id from departman d
```

```
    where upper(d.dept_ismi) like '%CRM%'
```

```
)
```

--Alıştırma-2

```
update iller i set bolge_adi =
```

```
(
```

```
    select bolge_adi from bolgeler b
```

```
    where b.bolge_kodu = i.bolge_kodu
```

```
);
```

--DELETE - Verileri Silme

```
delete from product p
where p.segment_id in
(
    select ps.id from product_segment ps
    where ps.segment = 'luxury'
)
```

```
delete from product p
where exists
(
    select 1 from product_segment ps
    where ps.segment = 'Luxury'
    and p.segment_id = ps.id
)
```

--COMMIT & ROLLBACK

```
insert into colors
values ('Red');
```

```
commit;
```

```
select * from cars;
```

```
update cars set discount = 1000
where id = 4;
```

```
rollback;
```

```
insert into colors values ('Pink');
insert into colors values ('Navy');
update colors set color='Light Blue'
where color='Blue'
```

```
commit or rollback;
```

```
select * from cars;
```

```
update cars set brand = 'Mercedes Benz'
where brand = 'Mercedes';
```

```
commit;
```

```
-----  
--SAVEPOINT Komutu  
-----
```

```
--Tüm segmentteki ürünlerin indirim oranı güncelleniyor
```

```
update product_segment set discount = discount + 0.08;
```

```
--Tüm ürünlerin indirim miktarları hesaplanarak net ücret güncelleniyor
```

```
update product p set net_price = price - price * (select s.discount from  
product_segment s where s.id = p.segment_id);
```

```
savepoint all_segment; --1.savepoint noktası
```

```
--Lüks segmentteki ürünlerin indirim oranı güncelleniyor
```

```
update product_segment set discount = discount + 0.15 where segment like '%Luxury%';
```

```
--Lüks segmentteki ürünlerin indirim miktarları hesaplanarak net ücret güncelleniyor
```

```
update product p set net_price = price - price * (select s.discount from  
product_segment s where s.id = p.segment_id)  
where p.segment_id in  
    (select ps.id from product_segment ps where ps.segment like '%Luxury%');
```

```
savepoint luxury_segment; --2.savepoint noktası
```

```
select sum(net_price) from product;
```

```
--Lüks olmayan segmentteki ürünlerin indirim oranı güncelleniyor
```

```
update product_segment set discount = discount + 0.05 where segment = 'Mass';
```

```
--Lüks olmayan segmentteki ürünlerin indirim miktarları hesaplanarak net ücret  
güncelleniyor
```

```
update product p set net_price = price - price * (select s.discount from  
product_segment s where s.id = p.segment_id)  
where p.segment_id in  
    (select ps.id from product_segment ps where ps.segment = 'Mass');
```

```
select sum(net_price) from product;
```

```
rollback;
```

```
-----  
--COMMIT - ROLLBACK - Satır Kilitleme  
-----
```

```
--Session-1
```

```
select * from cars;
```

```
update cars set price = 7500  
where id = 4;
```

```
--Session-2
```

```
select * from cars;
```

```
update cars set price = 8500  
where id = 4;
```

```
-----  
--FOR UPDATE  
-----
```

```
select * from personel where unvan = 'TEKNIKER' for update;  
13 rows selected
```

```
--Diğer kullanıcı ile
```

```
update egitim.personel set maas = 4000 where personel_id = 5020;
```

```
Commit;
```

--Tablo Oluşturma

```
CREATE TABLE EGITIM.CEZA_BILGI
(
  PERSONEL_ID          NUMBER,
  CEZA_ACIKLAMA         VARCHAR2(100 BYTE),
  BASLANGIC_TARIHI     DATE DEFAULT SYSDATE,
  BITIS_TARIHI          DATE
)
```

```
CREATE TABLE PERSONEL_EGITIM
(
  PERSONEL_ID          NUMBER,
  EGITIM_ADI           VARCHAR2(100),
  BASLANGIC_TARIHI     DATE,
  BITIS_TARIHI         DATE,
  EGITIM_TURU          INTEGER
)
```

```
create table customers
(
  customer_id number(8) not null,
  customer_name varchar2(100) not null,
  address varchar2(200),
  city varchar2(50),
  state varchar2(25),
  zip_code varchar2(10)
);
```

--Alt Sorgular ile Tablo Oluşturma

```
create table personel_uzman as
select personel_id, ad,
       soyad, maas,
       (maas+prim) hesaba_yatan
from personel
where unvan = 'UZMAN'
```

```
create table yonetici_yeni as
select yn.yonetici_id, ps.ad, ps.soyad,
       ps.unvan, kn.konum_adi
from yonetici yn, personel ps, konum kn
where ps.personel_id = yn.personel_id
       and ps.konum_id = kn.konum_id
```

--ALTER TABLE Komutları

--Kolon ekleme

alter table yoneticici **add** (sicil_no **number**);

--Kolon özelliği değiştirme

alter table yoneticici **modify** (unvan varchar2(150 byte));

--Kolona default değer verme

alter table yoneticici **modify** (yoneticici_id **default** 0);

--Kolon drop etme

alter table yoneticici **drop column** sicil_no;

--Kolon isimi değiştirme

alter table yoneticici **rename column** ad **to** adi;

--Tabloyu read-only moda geçirme

alter table yoneticici **read only**;

--Tablonun ismini değiştirme

alter table yoneticici **rename to** yoneticiler;

--Tabloya partition ekleme

alter table yoneticici **add partition** part2 **values** ('201005')

--ALTER TABLE Komutları - Comment

comment on table cars **is** 'Arabalara ait bilgiler bu tabloda tutulmaktadır';

comment on table personel **is** 'Bu tabloda, genel personel bilgileri yer almaktadır';

comment on column cars.brand **is** 'Araba markalarını belirtir';

comment on column product.net_price **is** 'Ürünlerin net ücretleri bu kolonda tutulmaktadır';

--ALTER TABLE Komutları - Drop Column

create table personel_temp **as select** * **from** personel;

alter table personel_temp **drop column** maas;

alter table personel_temp **drop column** ad;

alter table personel_temp **drop column** soyad;

--ALIŞTIRMALARIN CEVAPLARI

--Tablo Oluşturma

```
CREATE TABLE DERSLER
(
    DERS_ID NUMBER,
    DERS_ISMI VARCHAR2(50),
    DERS_ACIKLAMASI VARCHAR2(250),
    ILK_DERS_TARIHI DATE,
    DERS_KAPASITESI INTEGER DEFAULT 20,
    DERS_HOCASI VARCHAR2(70) DEFAULT 'Henüz atama yapılmamış'
)
```

--Alt Sorgular ile Tablo Oluşturma

```
create table customer_orders as
select
    c.customer_id, c.company_name, c.contact_name, c.city,
    o.order_date, o.freight, o.ship_city
from customers c, orders o
where c.customer_id = o.customer_id;
```

--ALTER TABLE Komutları

```
create table filmoyuncu
(
    filmid int,
    oyuncuismi varchar (70),
    oyuncusoyadi varchar (70)
);
```

--Kolon isimlerini kelimeler arasında alt tire (_) olacak şekilde değiştirin.

```
alter table filmoyuncu rename column filmid to film_id;
alter table filmoyuncu rename column oyuncuismi to oyuncu_ismi;
alter table filmoyuncu rename column oyuncusoyadi to oyuncu_soyadi;
```

--Doğum günü kolonu ekleyin.

```
alter table filmoyuncu add (dogum_tarihi date);
```

--250 karakterlik bir adres kolonu ekleyin

```
alter table filmoyuncu add (adres varchar2(250));
```

--Null olamayacak şekilde bir şehir kolonu ekleyin.

```
alter table filmoyuncu add (sehir varchar2(50) not null);
```

--Default değeri 'Warner Bros' olacak şekilde şirket kolonu ekleyin.

```
alter table filmoyuncu add (sirket varchar2(150) default 'Warner Bros');
```

--Adres kolonunun uzunluğunu 500 karaktere çıkartın.

```
alter table filmoyuncu modify (adres varchar2(500));
```

--Tablonun ismini film_oyuncu olarak değiştirin

```
alter table filmoyuncu rename to film_oyuncu;
```

--NOT NULL Constraint

CREATE TABLE PERSONEL_EGITIM

(
 PERSONEL_ID **NUMBER** **NOT NULL**,
 EGITIM_ADI VARCHAR2(100) **NOT NULL**,
 BASLANGIC **DATE**,
 BITIS **DATE**,
 EGITIM_TURU **INTEGER**
)

insert into personel_egitim
(personel_id, egitim_adi, baslangic, bitis, egitim_turu)
values (5555, null, sysdate+10, sysdate+15, 1)

--Tablo create edilirken:

create table kitap

(
 kitap_id **integer not null**,
 kitap_adi varchar2(120) **not null**,
 yazar_adi varchar2(100),
 isbn varchar2(40),
 basim_yili **integer**,
 sayfa_adedi **integer**
)
;

--Tablo varken:

alter table kitap **modify** yazar_adi varchar2(100) **not null**;

insert into kitap(kitap_id, kitap_adi, yazar_adi, isbn, basim_yili,
sayfa_adedi)
values(1, 'Oracle Öğreniyorum', null, null, 2023, 345);

--UNIQUE Constraint

```
CREATE TABLE PERSONEL_EGITIM
(
  PERSONEL_ID          NUMBER          NOT NULL,
  EGITIM_ADI           VARCHAR2(100) NOT NULL,
  BASLANGIC            DATE,
  BITIS                DATE,
  EGITIM_TURU          INTEGER,
  constraint pe_uc1 unique(personel_id, egitim_adi)
);
```

--Tablo varken:

```
create unique index pe_uc1 on personel_egitim (personel_id, egitim_adi)
```

--Tablo create edilirken:

```
create table kitap
(
  kitap_id integer not null,
  kitap_adi varchar2(120) not null unique,
  yazar_adi varchar2(100),
  isbn varchar2(40),
  basim_yili integer,
  sayfa_adi integer,
  unique(kitap_id),
  unique(kitap_adi, yazar_adi)
);
```

--Tablo varken:

```
alter table kitap add constraint uq_isbn unique(isbn);
```

--PRIMARY KEY Constraint

--Tablo create edilirken:

```
create table izin_turleri  
(  
    izin_turu integer primary key,  
    aciklama  varchar2(50 byte)  
);
```

--Tablo varken:

```
alter table izin_turleri  
    add constraint pk_izin_turu primary key (izin_turu);
```

```
-----  
--FOREIGN KEY Constraint  
-----
```

```
--Tablo create edilirken:
```

```
...  
constraint fk_konum foreign key (konum_id)  
           references konum (konum_id)  
...
```

```
--Tablo varken:
```

```
alter table personel add constraint fk_dept  
           foreign key (dept_id) references departman (dept_id)
```

```
-----  
create table musteri  
(  
    musteri_id number,  
    musteri_ismi varchar(255) not null  
);
```

```
create table yetkili  
(  
    yetkili_id number primary key,  
    musteri_id number,  
    yetkili_ismi varchar2(150) not null,  
    constraint fk_musteri  
           foreign key(musteri_id)  
           references musteri(musteri_id)  
);
```

```
alter table musteri  
add constraint pk_musteri primary key (musteri_id);
```

```
insert into musteri (musteri_id, musteri_ismi) values(1, 'Apple');  
insert into musteri (musteri_id, musteri_ismi) values(2, 'Google');  
insert into musteri (musteri_id, musteri_ismi) values(3, 'Amazon');
```

```
insert into yetkili (yetkili_id, musteri_id, yetkili_ismi) values(1, 1,  
'Alex');  
insert into yetkili (yetkili_id, musteri_id, yetkili_ismi) values(2, 2,  
'Michael');  
insert into yetkili (yetkili_id, musteri_id, yetkili_ismi) values(3, 3,  
'Jane');  
insert into yetkili (yetkili_id, musteri_id, yetkili_ismi) values(4, 3,  
'Judi');
```

--CHECK Constraint

--Tablo create edilirken:

```
create table personel
(
    sicil          number check(sicil > 0),
    maas           number,
    unvan          varchar2(50),
    bas_tarih      date,
    bit_tarih      date,
    constraint ck_maas1 check(maas >= 1000),
    constraint ck_maas2 check(maas between 1000 and 20000),
    constraint ck_unvan check(unvan in ('Uzman', 'Müdür')),
    check(bas_tarih < bit_tarih)
);
```

--Tablo varken:

```
alter table personel add constraint ck_maas1 check (maas > 1000)
```

--Constraint - Disable & Enable

```
alter table prm_iller add constraint fk_prm_iller
    foreign key (bolge_kodu)
    references prm_bolge(bolge_kodu) on delete cascade;
```

--Disable

```
alter table iller disable constraint fk_iller;
```

--Enable

```
alter table iller enable constraint fk_iller;
```

--Constraint Örnekleri

--Bölge tablosu oluşturuluyor

```
CREATE TABLE BOLGE
(
    BOLGE_KODU INTEGER PRIMARY KEY,
    BOLGE_ADI  VARCHAR2(100 BYTE) NOT NULL
);
```

--İl tablosu oluşturuluyor

```
CREATE TABLE IL
(
    IL_KODU      INTEGER NOT NULL,
    IL_ADI       VARCHAR2(50 BYTE) NOT NULL,
    BOLGE_KODU   INTEGER,
    CONSTRAINT FK_IL1 FOREIGN KEY (BOLGE_KODU) REFERENCES BOLGE(BOLGE_KODU)
);
```

--Bölgeler ekleniyor

```
INSERT INTO BOLGE VALUES(1, 'İÇ ANADOLU');
```

```
INSERT INTO BOLGE VALUES(1, 'MARMARA');
```

```
INSERT INTO BOLGE VALUES(2, '');
```

```
INSERT INTO BOLGE VALUES(2, 'MARMARA');
```

--İller ekleniyor

```
INSERT INTO IL VALUES(6, 'ANKARA', 1);
```

```
INSERT INTO IL VALUES(61, 'TRABZON', 5);
```

--Bölge siliniyor

```
DELETE FROM BOLGE WHERE BOLGE_KODU = 1;
```

```
-----  
--ALIŞTIRMALARIN CEVAPLARI  
-----
```

```
-----  
--PRIMARY KEY Constraint  
-----
```

```
alter table kitap  
add constraint pk_kitap primary key (kitap_id);
```

```
-----  
--FOREIGN KEY Constraint  
-----
```

```
alter table personel  
add constraint fk_konum_id foreign key (konum_id) references konum(konum_id)
```

```
-----  
--CHECK Constraint  
-----
```

```
alter table cars add (segment varchar2(1));
```

```
alter table cars  
add constraint ck_cars_segment check (segment in('A','B','C','D','E'));
```

```
insert into cars values(13,'Bugatti',340000,0,'S');
```

--View Oluşturma

```
create view vw_konum_bilgi as
select ad, soyad, semt, konum_adi
from personel pr, konum kn
where kn.konum_id = pr.konum_id;
```

```
create view vw_yoneticiler as
select yn.yoneticisi_id, ps.ad, ps.soyad,
       ps.unvan, pk.konum_adi,
       (ps.maas + nvl(ps.prim,0)) gelir
from yoneticisi yn, personel ps, konum pk
where ps.personel_id = yn.personel_id
      and ps.konum_id = pk.konum_id;
```

--View'ları Değiştirmek

```
create or replace view vw_konum_bilgi as
select ad || soyad adi_soyadi, semt, konum_adi, il_kodu
from personel pr, konum kn
where kn.konum_id = pr.konum_id
```

--Bağımlı View

```
create or replace view cust_order_prod_info as
select
    co.customer_id, co.company_name,
    co.order_date, co.freight, co.ship_city,
    p.product_name, p.unit_price
from
    cust_order_info co,
    order_details od, products p
where 1=1
      and co.order_id = od.order_id
      and p.product_id = od.product_id;
```



```
-----  
--Drop View  
-----
```

```
DROP VIEW CUST_ORDER_INFO;
```

```
-----  
--View Üzerinde DML İşlemleri  
-----
```

```
create view vw_uzman as  
select * from personel where unvan = 'UZMAN';  
  
select * from vw_uzman;  
  
update vw_uzman set konum_id = 5 where prim > 0;  
1 row updated  
  
select * from vw_uzman;
```

```
-----  
  
create view vw_product_grand_lux as  
select id, name, price from product  
where segment_id = 1;  
  
update vw_product_grand_lux  
set price = price * 1.3;  
  
delete from vw_product_grand_lux  
where id = 11;  
  
insert into vw_product_grand_lux(id, name, price)  
values(21, 'Air Cleaner', 1200);
```

--View Üzerinde DML İşlemleri – With Read Only

```
create or replace view vw_uzman as
select * from personel where unvan = 'UZMAN'
with read only;

update vw_uzman set konum_id = 5 where prim > 0;
```

--View Üzerinde DML İşlemleri – With Check Option

```
create or replace view vw_uzman as
select * from personel where unvan = 'UZMAN';

update vw_uzman set unvan = 'ŞEF';
15 rows updated.

select * from vw_uzman;

create or replace view vw_uzman as
select * from personel where unvan = 'UZMAN'
with check option;

update vw_uzman set unvan = 'ŞEF';
```

--ALIŞTIRMALARIN CEVAPLARI

--View Oluşturma

```
create view cust_order_info as  
select  
    c.customer_id, c.company_name, c.contact_name,  
    o.order_id, o.order_date, o.freight, o.ship_city  
from customers c, orders o  
where c.customer_id = o.customer_id
```

```
-----  
--Index Oluşturma  
-----
```

```
CREATE INDEX PERSONEL_NI1 ON PERSONEL(PERSONEL_ID);
```

```
-----  
CREATE INDEX PI_NI1 ON PERSONEL_IZINLERI(PERSONEL_ID, IZIN_TURU)
```

```
-----  
--Index Üzerinden Verilere Erişme  
-----
```

```
select * from personel;
```

```
select * from personel where ad = 'SERKAN';
```

```
select * from personel where personel_id = 1000;
```

```
-----  
--Index Oluşturma  
-----
```

```
alter table personel add (email varchar2(150));
```

```
select  
    translate(lower(replace(ad, ' ', '')), '1öüçşğ', 'ioucsg') || '.' ||  
    translate(lower(replace(soyad, ' ', '')), '1öüçşğ', 'ioucsg') ||  
'@dbhunter.net' as email  
from personel;
```

```
update personel set email =  
    translate(lower(replace(ad, ' ', '')), '1öüçşğ', 'ioucsg') || '.' ||  
    translate(lower(replace(soyad, ' ', '')), '1öüçşğ', 'ioucsg') ||  
'@dbhunter.net';
```

```
--Unique index oluşturmaya çalışıyoruz fakat hata alacağız
```

```
create unique index uq_personel_email on personel(email);
```

```
--Tekrarlayan email adreslerinden birini güncelliyoruz
```

```
update personel set email = 'servet.turan2@dbhunter.net' where personel_id  
= 5084;
```

```
explain plan for
select
    c.customer_id, c.company_name, c.contact_name,
    o.order_id, o.order_date, o.freight, o.ship_city
from customers c, orders o
where c.customer_id = o.customer_id;
```

```
select plan_table_output
from table(dbms_xplan.display());
```

```
select * from customers where customer_id = 'PARIS';
```

```
-----
--Function Based Index Oluşturma
-----
```

```
CREATE INDEX PERSONEL_NI2 ON PERSONEL(UPPER(AD));
```

```
select * from personel where upper(ad) = 'SELEN'
```

```
-----  
--ALIŞTIRMALARIN CEVAPLARI  
-----
```

```
-----  
--Index Oluşturma  
-----
```

```
--1. index
```

```
create index ix_personel_id on personel(personel_id);
```

```
--2. index
```

```
create index uq_emp_city_postal on employees (city, postal_code);
```

```
--3. index
```

```
create index uq_student_name_course on student (name, course_name);
```

```
--4. index
```

```
create index ix_mudurluk_il on mudurluk(lower(il_adi));
```

--SEQUENCE – Oluşturma

```
create sequence sq_konum
    increment by 1
    start with 10
    maxvalue 9999
    nocache
    nocycle;
```

--SEQUENCE Kullanımı

```
insert into konum(konum_id, konum_adi, il_kodu)
values(sq_konum.nextval, 'Mecidiyeköy', 34);

insert into konum(konum_id, konum_adi , il_kodu)
values(sq_konum.nextval, 'Gölbasi', 6);

select * from konum order by konum_id;
```

--SEQUENCE' i Değiştirme

```
ALTER SEQUENCE SEQ_KONUM
    INCREMENT BY 11
    MAXVALUE 10000
    NOCACHE
    NOCYCLE;
```

--SYNONYM

```
create synonym sube for mudurluk_sube;
```

```
select * from sube;
```

```
--Egitim2 kullanisiyla baglanalim ve ayni select'i calistiralim
select * from sube; !!!
```

--SYNONYM – Public

```
drop synonym sube;
```

```
create public synonym sube for prm_mudurluk_sube;
```

```
--Diğer kullanıcıyla bağlanalım ve aynı SQL'i çalıştıralım
```

```
select * from sube;
```

```
-----  
--DROP  
-----
```

```
drop synonym mudurluk
```

```
drop public synonym sube
```

```
drop index personel_ni1
```

```
drop table abone purge
```

```
drop constraint personel_fk1
```

```
--ALIŞTIRMALARIN CEVAPLARI
```

```
--SEQUENCE - Oluşturma
```

```
--Alıştırma-1
```

```
create sequence sq_geriye_dogru
  increment by -2
  start with 100
  maxvalue 100;
```

```
--Alıştırma-2
```

```
alter table colors add(color_id number);
```

```
create sequence sq_colors
  increment by 1
  start with 1
  nomaxvalue;
```

```
update colors set color_id = sq_colors.nextval;
```

```
-----  
--TIME ZONES  
-----
```

```
ALTER SESSION SET TIME_ZONE = '+03:00'  
ALTER SESSION SET TIME_ZONE = dbtimezone  
ALTER SESSION SET TIME_ZONE = local  
ALTER SESSION SET TIME_ZONE = 'Turkey'
```

```
-----  
--ŞİMDİKİ ZAMAN  
-----
```

```
SELECT CURRENT_DATE, CURRENT_TIMESTAMP, LOCALTIMESTAMP FROM DUAL;  
ALTER SESSION SET TIME_ZONE = '-3:00';  
SELECT DBTIMEZONE, SESSIONTIMEZONE FROM DUAL;
```

```
-----  
--TIME_STAMP Karşılaştırma  
-----
```

```
create table zaman_ornek  
(  
  zaman1 timestamp with time zone,  
  zaman2 timestamp with local time zone,  
  zaman3 date  
);  
  
insert into zaman_ornek  
values (current_date, current_timestamp + 2, sysdate);  
  
select * from zaman_ornek;
```

--INTERVAL YEAR TO MONTH

```
create table uyarici
(
    uyarici_id number,
    uyarici_araligi interval year(3) to month
);

insert into uyarici values (1, interval '6' month);
insert into uyarici values (2, interval '-3' month);
insert into uyarici values (3, interval '100' year(3));
insert into uyarici values (4, '50-10');
insert into uyarici values (5, '-1-2');

select trunc(sysdate) simdiki_zaman,
       uyarici_araligi,
       trunc(sysdate + uyarici_araligi) uyari_zamani
from uyarici;
```

--INTERVAL DAY TO SECOND

```
create table sure_asimi
(
    sure_id number,
    sure_araligi interval day(2) to second
);

insert into sure_asimi values (1, '90 00:00:00');
insert into sure_asimi values (2, interval '6 03:30:16' day to second);

select sysdate, sure_araligi, (sysdate + sure_araligi) sure_zamani
from sure_asimi;
```

--TO_YMINTERVAL

```
select ad, giris_tarihi,
       (giris_tarihi + to_yminterval('02-03')) int_tarih
from personel
where giris_tarihi is not null
      and unvan = 'MÜDÜR';
```

--TO_DSINTERVAL

```
select ad,
       to_char(giris_tarihi, 'dd.mm.yy hh24:mi:ss') giris1,
       to_char((giris_tarihi + to_dsinterval('10 10:10:10')), 'dd.mm.yy
hh24:mi:ss') giris2
from personel
where giris_tarihi is not null;
```

--INTERVAL Veri Tipleri - Örnek

```
create table interval_ornek
(
  sure interval year(3) to month
);

insert into interval_ornek values('10-02');

select trunc(sysdate), sure,
       trunc(sysdate + sure) sonuc1,
       trunc(sysdate + to_ymininterval('10-02')) sonuc2,
       trunc(add_months(sysdate, 122)) sonuc3
from interval_ornek;
```

--EXTRACT

```
select ad, soyad, giris_tarihi,
       extract (year from giris_tarihi) yil,
       extract (month from giris_tarihi) ay,
       extract (day from giris_tarihi) gun
from personel;
```

--Başka Bir Tablodan Satır Kopyalama

```
insert into ceza_bilgi
(
    personel_id, ceza_aciklama,
    baslangic_tarihi, bitis_tarihi
)
select
    personel_id, 'Eylem Yapmak',
    sysdate, sysdate+10
from personel
where unvan = 'UZMAN'
```

```
insert into personel_izinleri(
    personel_id, izin_turu,
    izin_bas_tarihi, izin_bit_tarihi)
select
    personel_id, 11,
    '10.03.2020', '12.03.2020'
from personel
where unvan = 'MÜDÜR'
```

```
create table personel_bordro
(
    personel_id number,
    maas number,
    prim number
);

insert into personel_bordro(personel_id, maas, prim)
select personel_id, maas, prim
from personel;
```

```
create table uk_emp_orders as
select
    e.employee_id, e.first_name , e.last_name,
    o.order_date, o.freight, o.ship_city
from employees e, orders o
where e.country = 'UK'
and e.employee_id = o.employee_id
```

```
--ŞARTSIZ INSERT ALL
```

```
insert all
  into personel_sicil values (personel_id, ad, soyad)
  into personel_maas values (personel_id, maas, prim)
select ad, soyad, personel_id, maas, prim
from personel
where unvan = 'UZMAN';
```

```
--ŞARTLI INSERT ALL
```

```
insert all
  when unvan = 'UZMAN' then
    into personel_uzman values (personel_id, ad, soyad, maas, unvan)
  when unvan in ('MÜDÜR', 'GRUP MÜDÜRÜ') then
    into personel_yonetici values (personel_id, ad, soyad, maas, unvan)
select ad, soyad, personel_id, maas, prim, unvan
from personel;
```

```
--ŞARTLI INSERT FIRST
```

```
insert first
  when maas <= 3000 then
    into dusuk_maas values (personel_id, ad, soyad, maas)
  when maas between 3000 and 6000 then
    into orta_maas values (personel_id, ad, soyad, maas)
  else
    into yuksek_maas values (personel_id, ad, soyad, maas)
select personel_id, ad, soyad, maas
from personel
```

```
insert first
  when unvan = 'UZMAN' then
    into personel_uzman values (personel_id, ad, soyad, maas, unvan)
  when unvan in ('MÜDÜR', 'GRUP MÜDÜRÜ') then
    into personel_yonetici values (personel_id, ad, soyad, maas, unvan)
  when unvan in ('UZMAN', 'MÜHENDİS') then
    into personel_diger values (personel_id, ad, soyad, maas, unvan)
select ad, soyad, personel_id, maas, prim, unvan
from personel;
```

--DIKEY INSERT

```
create table bilgiler
(
    personel_id number,
    kod integer,
    aciklama varchar2(100),
    deger varchar2(100)
);

insert all
into bilgiler values(personel_id, 1, 'Adı-Soyadı', ad||' '||Soyad)
into bilgiler values(personel_id, 2, 'Unvan', unvan)
into bilgiler values(personel_id, 3, 'Maaş', to_char(maas))
into bilgiler values(personel_id, 4, 'Semt', semt)
into bilgiler values(personel_id, 5, 'Giriş Tairihi',
to_char(giris_tarihi, 'dd.mm.yyyy'))
select * from personel;
```

--MERGE

--Sadece uzman, mühendislerin olduğu bir personel tablosu oluşturuluyor

```
create table prs1 as
select personel_id, ad, soyad, maas, unvan
from personel where unvan in ('MÜHENDİS', 'UZMAN');
```

--Sadece müdür, uzman, mühendislerin olduğu bir personel tablosu oluşturuluyor

```
create table prs2 as
select personel_id, ad, soyad, maas, unvan
from personel where unvan in ('MÜHENDİS', 'UZMAN', 'MÜDÜR');
```

--Müdür harici personelin maaşına %5 zam yapılıyor

```
update prs2 set maas = maas*1.05 where unvan != 'MÜDÜR';
```

--2 tablo prs1 üzerinde merge ediliyor

```
merge into prs1 p1
using prs2 p2
on (p1.personel_id = p2.personel_id)
when matched then
    update set p1.maas = p2.maas
    delete where maas > 5000
when not matched then
    insert (personel_id, ad, soyad, maas, unvan)
    values (p2.personel_id, p2.ad, p2.soyad, p2.maas, p2.unvan);
```

--ALIŞTIRMALARIN CEVAPLARI

--Başka Bir Tablodan Satır Kopyalama

```
insert into basket_c (fruit_c)
select fruit_a
from basket_a
union
select fruit_b
from basket_b;
```

--ŞARTLI INSERT ALL

```
insert into personel_yoneticisi
select personel_id, ad, soyad, maas, unvan
from personel
where unvan in ('MÜDÜR', 'GRUP MÜDÜRÜ');
```