

KARMAŞIK SORGULAR (2)

ALT SORGULAR (İÇ İÇE SORGULAR)

- ❑ İç içe yazılan sorgular, içten dışa veya aşağıdan yukarıya doğru çalıştırılarak, her sorgunun sonucu diğer sorguya girdi olarak verilir. En sonunda, en dıştaki sorgunun sonucu alınır.
- ❑ Özellikle birden fazla tabloya aynı anda erişilmesi gereken durumlarda alt sorgu ile sorgulamak kolay ve hızlı sonuçlar üretebilir.

Alt Sorgu Özellikleri

- Alt sorgular parantez içinde yazılır.
- Order By ve Group By içerisinde alt sorgu kullanılmaz.
- Alt sorgulara kolaylık olması açısından takma isimler verilebilir.
- Alt sorgu = ifadesi ile karşılaştırılmışsa alt sorgudan dönen değer tek olmalıdır.
- Eğer birden fazla değer dönüyorsa **IN** veya **NOT IN** kullanılmalıdır.

SELECT İfadesi için Alt Sorgu

```
Select sutunlar From tablo_adi  
where koşul_ifadesi (
```

```
Select kaynak_sutun_adi  
From kaynak_tablo_adi
```

```
)
```

ÖRNEK

Kerem YILMAZ ile aynı bölümde çalışan personelleri listeleyelim.

PERSONEL

personelNo	ad_soyad	gorev	maas	bNo
1	Ali SAĞIR	Avukat	4800	3
2	Fatih GÜRAL	Mühendis	5600	2
3	Hatice YILMAZ	Muhasebeci	4500	4
4	Mehmet YAŞAR	Avukat	8000	3
5	Fadime ERDÖNMEZ	Satış Temsilcisi	5400	1
6	Levent SERT	Teknisyen	4300	2
7	Kerem YILMAZ	Mühendis	6700	2
8	Menekşe KAYIM	Satış Temsilcisi	7650	1

BÖLÜM

bNo	bolum_adi
1	Müşteri Hizmetleri
2	Teknik Servis
3	Hukuk Müşavirliği
4	Tahakkuk Servisi
5	Lojistik
6	Güvenlik

Normalde SQL sorgusu aşağıdaki gibi yazılabilir.

SELECT * FROM personel WHERE bNo=2;

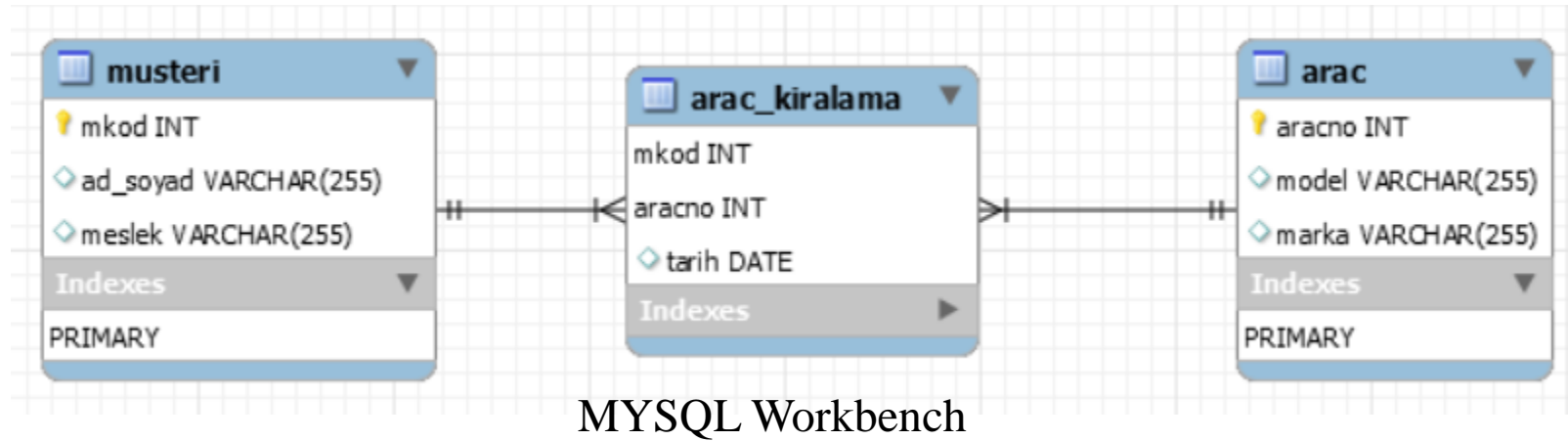
personelNo	ad_soyad	gorev	maas	bNo
2	Fatih GÜRAL	Mühendis	5600	2
6	Levent SERT	Teknisyen	4300	2
7	Kerem YILMAZ	Mühendis	6700	2

Fakat bir kişinin her zaman bölüm numarasını bilemeyiz. Dolayısıyla aşağıdaki gibi iç içe bir sorguya ihtiyacımız olacaktır.

SELECT * FROM personel WHERE bNo=(SELECT bNo FROM personel WHERE ad_soyad="Kerem YILMAZ");

ÖRNEK (2)

Örneğin bir araç kiralama acentesi ile ilgili Varlık-İlişki (E-R) diyagramının bir kısmı aşağıda gösterilmiştir.



- **Müşteri** ve **araç** varlıkları arasındaki çoğa-çok ilişkisinin sağlanması için **arac_kiralama** adında bir geçiş (ara) bir tablo oluşturulmuştur.
- Bu tabloda bir müşterinin kiralamış olduğu **aracın markasını öğrenmek** için **araç** ve **arac_kiralama** tabloları üzerinden sorgu yapılması gerekir.

ÖRNEK (2)

- Tablolara aşağıdaki gibi örnek kayıtlar eklenmiştir.

Müşteri

mkod	ad_soyad	meslek
1	Ali BALCI	Öğretmen
2	Necdet GÜRSES	Akademisyen
3	Nazım ÖZEL	Mühendis
4	Asiye USLU	Öğretmen
5	Murat EKİN	Mühendis

Araç Kiralama

mkod	aracno	tarih	ucret
1	103	2020-01-04	300
1	105	2020-02-17	600
2	101	2019-05-15	300
2	102	2020-01-04	750
3	101	2020-02-17	250
3	105	2020-03-11	350
2	101	2020-03-11	350
4	101	2020-02-12	340
4	101	2020-03-12	340
4	101	2020-03-18	340

Araç

aracno	model	marka
101	2009	bmw
102	2011	mercedes
103	2019	opel
104	2020	fiat
105	2020	Honda

Örneğin 1 numaralı müşterinin kiralamış olduğu araçların markası listelensin.

Sorgu: `SELECT marka FROM arac WHERE aracno IN (SELECT aracno FROM arac_kiralama WHERE mkod=1);`

Araç

marka
opel
Honda

Not: Alt sorguda geriye dönen değerlerde araç numarası birden fazla olacaktır.

Bu nedenle aracno'dan sonra = yerine **in** ifadesi kullanılmıştır.

ÖRNEK (2)

Sorguyu biraz daha genişletebiliriz. Örneğin müşterinin numarasını bilinmiyorsa yeni bir alt sorgu yazarız. Ali BALCI'nın kiralamış olduğu araçların markalarını listelemek için aşağıdaki sorgu yazılır.

```
SELECT marka FROM arac WHERE aracno IN  
(SELECT aracno FROM arac_kiralama  
WHERE mkod= (SELECT mkod FROM musteri WHERE  
ad_soyad='Ali BALCI')  
);
```

Araç

marka
opel
Honda

ÖRNEK (2)

Ali BALCI'nın kiralamamış olduğu araçların markaları listelemek için aşağıdaki sorgu yazılır.

```
SELECT marka FROM arac WHERE aracno NOT IN  
(SELECT aracno FROM arac_kiralama  
WHERE mkod= (SELECT mkod FROM musteri WHERE  
ad_soyad='Ali BALCI')  
);
```

Araç

marka
bmw
mercedes
fiat

ÖRNEK (3)

ORDER BY, GROUP BY gibi ifadeler, ana sorgularda kullanılabilir. Fakat alt sorgularda kullanılamaz.

Örneğin Nazım ÖZEL adlı müşterinin araç kiralama ücretleri büyükten küçüğe doğru listeleyen sorgu aşağıdadır.

```
SELECT * FROM arac_kiralama WHERE mkod=(SELECT mkod  
FROM musteri WHERE ad_soyad='Nazım ÖZEL')  
ORDER BY ucret DESC;
```

Araç Kiralama

mkod	aracno	tarih	ucret
3	105	2020-03-11	350
3	101	2020-02-17	250

ÖRNEK (4)

Örneğin Nazım ÖZEL adlı müşterinin **hangi tarihte hangi marka, model araba kiralamış ve ne kadar ücret ödemiş olduğu** bilgilerine aşağıdaki SQL sorgusu ile ulaşılabilir.

```
SELECT a.model,a.marka,k.ucret,k.tarih FROM arac_kiralama as k INNER JOIN  
arac as a ON k.aracno= a.aracno AND k.mkod=(SELECT mkod FROM musteri  
WHERE ad_soyad='Nazım ÖZEL' ) ;
```

- **arac_kiralama** tablosunda yalnızca **aracno** var. **Arac** tablosunda aracın **model** ve **marka** bilgileri mevcut. Bu nedenle iki tablodaki ortak alanlar **inner join** ile birleştirilmiştir.
- Daha sonra **arac_kiralama** tablosunda **mkod** bilgisi ile 'Nazım ÖZEL'in elde edilen **mkod** bilgisi eşleştirilmiştir.

Araç Kiralama

model	marka	ucret	tarih
2009	bmw	250	2020-02-17
2020	Honda	350	2020-03-11

ÖRNEK (5)

BMW markalı aracı kiralayan kişiler listelenmesi istensin.

- BMW markalı aracın araç numarası **arac** tablosundan bulunur.
- Bu aracı kimler kiralamış **arac_kiralama** tablosundan erişilir.
- Bu kişilerin bilgilerine ise **müşteri** tablosundan erişilecektir.

SORGU

```
SELECT * FROM musteri WHERE mkod IN( select mkod from arac_kiralama  
WHERE aracno=(select aracno FROM arac WHERE marka='bmw') );
```

	mkod	ad_soyad	meslek
	2	Necdet GÜRSES	Akademisyen
	3	Nazım ÖZEL	Mühendis
	4	Asiye USLU	Öğretmen

ÖRNEK (5)

Bir başka örnekte ise BMW markalı araç kimler tarafından kaç kez kiralanmış ve kiralayan her bir kişi toplam ne kadar ücret ödemiş listelenmesi istensin.

- Her bir müşterinin BMW markalı araç için kaç kez kiralandığı ve toplam ne kadar ücret ödendiği hesaplanması için **arac_kiralama** tablosunda müşteri koduna göre gruplandırma işlemi yapılmalıdır.
- Müşterinin bilgilerine erişilmesi de için müşteri ve arac_kiralama tablosu ortak alan olan **mkod** alanına göre birleştirilmelidir.

❑SORGU :

```
SELECT m.mkod, m.ad_soyad, count(k.mkod) as kiralama_sayisi, sum(k.ucret) as  
tpl_ucret FROM musteriler as m INNER JOIN arac_kiralama as k ON m.mkod=k.mkod  
AND k.aracno=(select aracno FROM arac WHERE marka='bmw') group by k.mkod
```

mkod	ad_soyad	kiralama_sayisi	tpl_ucret
2	Necdet GÜRSES	2	650
3	Nazım ÖZEL	1	250
4	Asiye USLU	3	1020

Alt Sorgu Örneği – Update (1)

Müşteri Tablosunda 5 numaralı müşterinin soyadını ‘EKİNCİ’ olarak değiştirilsin.

```
Update müşteri SET ad_soyad= ‘Murat EKİNCİ’  
WHERE mkod=7
```

Mkod’unu bilindiği durumlarda bu güncelleme geçerli olmaktadır.

Alt Sorgu Örneği – Update (2)

- Müşteri kodu/numarasını alt sorguyla belirlenerek güncelleme işlemi gerçekleştirilebilir. Bu durumda aşağıdaki gibi bir sorgu yazılır.

```
UPDATE musteriler, ( SELECT m.mkod FROM musteriler as m WHERE  
m.ad_soyad='Murat EKİN' ) as mus  
  
SET musteriler.ad_soyad= 'Murat EKİNCİ'  
  
WHERE musteriler.mkod=mus.mkod
```

NOT: Yukarıdaki sorguda görüldüğü gibi alt sorgu update SET kısmından önce yazılır.

Alt Sorgu Örneği – Update (3)

- Bir başka örnek vermek gerekirse 2019 modelden düşük araçların kiralama ücretleri %10 oranında indirim yapılarak tüm müşterilere yansıtılsın.

```
UPDATE arac_kiralama as k,
```

```
(SELECT * FROM arac_kiralama WHERE aracno IN
```

```
(SELECT aracno FROM arac WHERE model<2019 )) as a
```

```
SET k.ucret= k.ucret*0.9 WHERE k.aracno=a.aracno
```

Araç Kiralama

mkod	aracno	tarih	ucret
1	103	2020-01-04	300
1	105	2020-02-17	600
2	101	2019-05-15	270
2	102	2020-01-04	675
3	101	2020-02-17	225
3	105	2020-03-11	350
2	101	2020-03-11	315
4	101	2020-02-12	306
4	101	2020-03-12	306
4	101	2020-03-18	306

2019 modelin altında olan araçların arac numaraları araç tablosundan alınıp arac_kiralama tablosu ile eşleştirildi. Daha sonra eşleşen kayıtlarda ücret sütunu %10 oranında düşürüldü.

Alt Sorgu Örneği - Delete

Örneğin Asiye USLU'nun araç kiralama kayıtlarını silinmesi aşağıdaki sorgu ile yapılır.

```
DELETE FROM arac_kiralama WHERE mkod=(SELECT mkod  
FROM musteri WHERE ad_soyad='Asiye USLU')
```

Araç Kiralama

mkod	aracno	tarih	ucret
1	103	2020-01-04	300
1	105	2020-02-17	600
2	101	2019-05-15	270
2	102	2020-01-04	675
3	101	2020-02-17	225
3	105	2020-03-11	350
2	101	2020-03-11	315

Birden Fazla Tabloda Kayıt Silinmesi

- MySQL'de veritabanları **INNODB** ve yabancıl anahtar alan kısıtlamalarında **ON DELETE CASCADE** olarak tanımlandığında bir tabloda bir kayıt silindiğinde o tablo ile ilgili olan diğer tablolardaki kayıtlar otomatik olarak silinmektedir.
- Veritabanları MyISAM olarak tanımlanırsa bir tablodan kayıt silinince diğer tabloda kayıt silinmesi için ekstra SQL sorguları yapmak gerekmektedir.

Birden Fazla Tabloda Kayıt Silinmesi (2)

MYISAM veritabanı yapısına sahip veri tabanlarında birden fazla tablodan kayıt silmek için iki farklı sorgulama biçimi kullanılmaktadır.

1. **DELETE** *t1, t2* **FROM** *tablo1* **AS** *t1* **INNER JOIN** *tablo2* **AS** *t2*
WHERE *t1.id=t2.id*;

2. **DELETE FROM** *t1, t2* **USING** *tablo1* **AS** *t1* **INNER JOIN** *tablo2* **AS** *t2*
WHERE *t1.id=t2.id*;

Birden Fazla Tabloda Kayıt Silinmesi -Örnek

- Örneğin aşağıda verilen bölüm tablosunda 1 numaralı bölüm kaldırılınsın ve bu bölümdeki çalışan tüm personeller de çalışanlar listesinden çıkarılsın.

PERSONEL

pno	ad_soyad	bno
102	Mustafa TEMEL	1
101	Kadir YILMAZ	1
104	Sude YILMAZ	2
103	Osman KAYIM	2
105	Yağmur SERT	3
106	Duru YILMAZ	3

BÖLÜM

bno	ad
1	Teknik Servis
2	Lojistik
3	Müşteri Hizmetleri

SORGU: DELETE t1, t2 FROM bolum as t1 INNER JOIN personel as t2 WHERE t1.bno=t2.bno AND t1.bno=1;

pno	ad_soyad	bno
104	Sude YILMAZ	2
103	Osman KAYIM	2
105	Yağmur SERT	3
106	Duru YILMAZ	3

bno	ad
2	Lojistik
3	Müşteri Hizmetleri

Birden Fazla Tabloda Kayıt Silinmesi –Örnek (2)

- Lojistik bölümü ve bu bölümle ilişkili tüm personelleri tablolardan kaldırılсын.

PERSONEL

pno	ad_soyad	bno
104	Sude YILMAZ	2
103	Osman KAYIM	2
105	Yağmur SERT	3
106	Duru YILMAZ	3

BÖLÜM

bno	ad
2	Lojistik
3	Müşteri Hizmetleri

SORGU: DELETE t1, t2 FROM bolum AS t1 INNER JOIN personel AS t2 , (SELECT bno FROM bolum WHERE ad='lojistik') AS t3 WHERE t1.bno=t2.bno AND t2.bno=t3.bno;

pno	ad_soyad	bno
105	Yağmur SERT	3
106	Duru YILMAZ	3

bno	ad
3	Müşteri Hizmetleri

Kaynakça

- <https://dev.mysql.com/doc/refman/8.0/en/sql-data-manipulation-statements.html>