

# Veri Tabanı - 1

5. Hafta Dersi



# Dersin Hedefleri

## □ İlişkisel Cebir

- Seçim (Selection)
- Atma (Projection)
- Kartezyen Çarpım (Cross-Product)
- Birleşim (Union)
- Küme farkı (Set-difference)
- Kesişim (Intersection)
- Bölme (Division)
- Birleştirme (Join)
- Yeniden Adlandırma (Renaming)

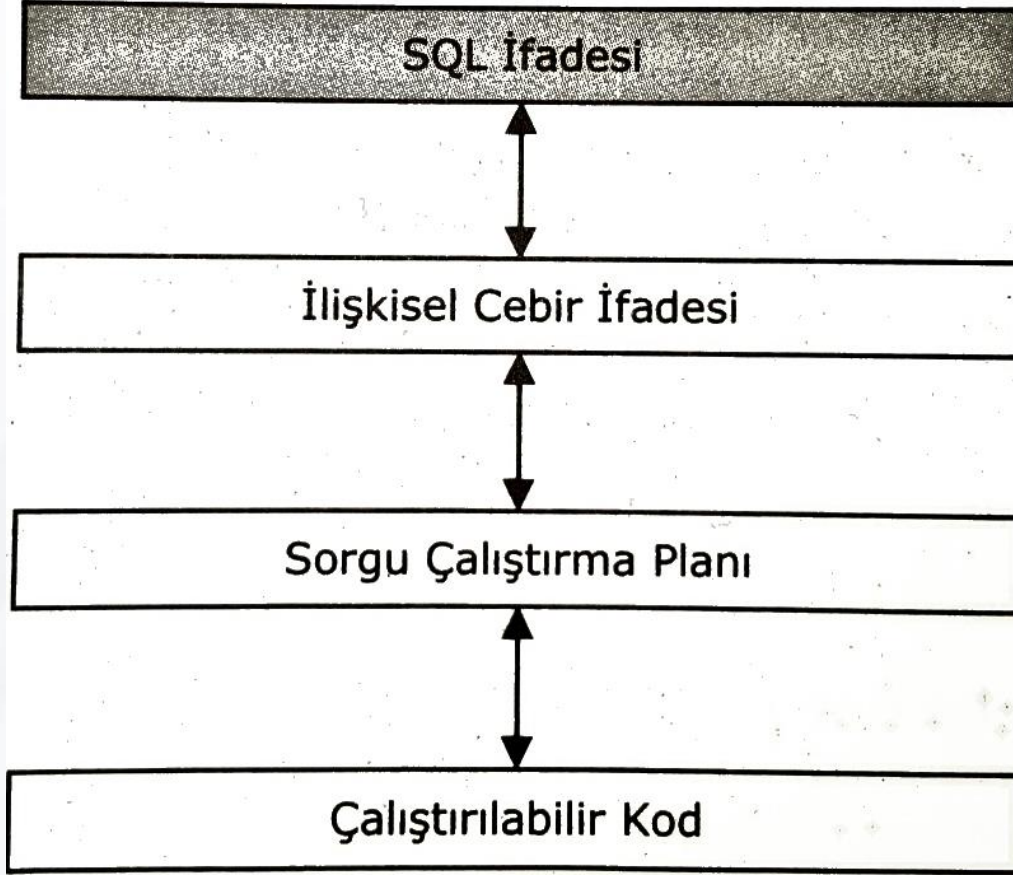
# İlişkisel Cebir

- İlişkisel veri tabanları üzerinde yapılan sorgulama işlemlerinin matematiksel olarak modellenmesi ve matematiksel işlemler ile çözülebilmesi için kullanılır.

# İlişkisel Cebir

- İlişkisel cebir ilişkisel modeli ile birlikte kullanılan veri tabanı sorgulama dilidir.
- SQL den farklı olarak herhangi bir yorumlayıcıya veya derleyiciye gerek yoktur
- Sorgular çeşitli operatörlerin birleşiminden meydana gelir.

# İlişkisel Cebir



- **Veri tabanı yönetim sistemi yazılan ifadeleri veri tabanına uygulamadan önce ilişkisel cebir işlemine çevirmektedir.**

# İlişkisel Cebir

- İlişkisel cebir konusunu anlamanız ileride kullanacağımız SQL sorgulama dilinin mantığını kavramakta size yardımcı olacaktır.

# İlişkisel Cebir

- Seçim (Selection)
- Atma (Projection)
- Kartezyen Çarpım (Cross-Product)
- Birleşim (Union)
- Küme farkı (Set-difference)
- Kesişim (Intersection)
- Bölme (Division)
- Birleştirme (Join)
- Yeniden Adlandırma (Renaming)

# Seçme İşlemi ( $\sigma$ )

- Belirli bir ilişkiden bazı kayıtların seçilerek ortaya konulması işlemidir.  $\sigma$  işareti ile gösterilmektedir.
- Kullanımı:
- $\sigma$  (Seçim Kriteri)(TABLO ADI)
- Seçim işleminde karşılaştırma işleçleri kullanılır.
- =, <, >,  $\neq$ ,  $\leq$ ,  $\geq$
- Ayrıca mantıksal operatörler olan ve için  $\wedge$  veya için v kullanılır.



# Seçme İşlemi ( $\sigma$ )

PERSONEL

<u>PersonelNo</u>	<u>ŞubeNo</u>	<u>ŞubeAdres</u>	<u>İsim</u>	<u>Pozisyon</u>	<u>ÇalışmaSaati</u>
S4153	B002	İSTANBUL	AYŞE DEMİR	ASİSTAN	16
S4554	B004	İZMİR	SONER SARI	ASİSTAN	13
S4612	B002	İSTANBUL	MEHMET ÖDER	UZMAN	12
S4612	B004	İZMİR	KEVSER BİLİR	UZMAN	15

- Örnek: Personel tablosunda izmir şubesinde çalışan personelleri listeleyiniz.

•  $\sigma$ ŞubeAdres = İZMİR (PERSONEL)

# Seçme İşlemi ( $\sigma$ )

## PERSONEL

<u>PersonelNo</u>	<u>ŞubeNo</u>	<u>ŞubeAdres</u>	<u>İsim</u>	<u>Pozisyon</u>	<u>ÇalışmaSaati</u>
S4554	B004	İZMİR	SONER SARI	ASİSTAN	13
S4612	B004	İZMİR	KEVSER BİLİR	UZMAN	15

- $\sigma$  ŞubeAdres = İZMİR (PERSONEL)

# Seçme İşlemi ( $\sigma$ )

ÜRÜNLER

<u>Urun kod</u>	<u>Urun adi</u>	Marka	Fiyat
U001	URUN1	ABC	350
U002	URUN2	XYZ	750
U003	URUN3	XYZ	900
U004	URUN3	ABC	850
U005	URUN2	ABC	850

- Örnek: Markası XYZ ve fiyatı 300' den fazla olan ürünleri ilişkisel cebir ifadesiyle listeleyiniz.

$$\sigma_{\text{Marka} = \text{XYZ} \wedge \text{Fiyat} > 300} (\text{ÜRÜNLER})$$

# Seçme İşlemi ( $\sigma$ )

ÜRÜNLER

<u>Urun_kod</u>	<u>Urun_adi</u>	Marka	Fiyat
U002	URUN2	XYZ	750
U003	URUN3	XYZ	900

$\sigma_{\text{Marka} = \text{XYZ} \wedge \text{Fiyat} > 300}$  (ÜRÜNLER)

# Atma (Projection) $\Pi$

- Belirli bir ilişkiden sadece bazı sütunları atmak için kullanılır.
- Seçim işleminden dönen sonuçlar ile de kullanılabilir.
- Kullanımı:
- $\Pi$  sütun isimleri (TABLO ADI)

# Atma (Projection) $\Pi$

ÜRÜNLER

<u>Urun kod</u>	<u>Urun adi</u>	Marka	Fiyat
U001	URUN1	ABC	350
U002	URUN2	XYZ	750
U003	URUN3	XYZ	900
U004	URUN3	ABC	850
U005	URUN2	ABC	850

ÖRNEK: Ürünler tablosundaki kayıtların sadece ürün isimleri ve markalarını listeleyen ilişkisel cebir ifadesini yazınız.

$\Pi$ Urun\_adi, Marka(ÜRÜNLER)

# Atma (Projection) $\Pi$

ÜRÜNLER

<u>Urun_adi</u>	Marka
URUN1	ABC
URUN2	XYZ
URUN3	XYZ
URUN3	ABC
URUN2	ABC

$\Pi$  Urun\_adi, Marka(ÜRÜNLER)

# Atma (Projection) $\Pi$

## ÜRÜNLER

<u>Urun kod</u>	<u>Urun adi</u>	Marka	Fiyat
U001	URUN1	ABC	350
U002	URUN2	XYZ	750
U003	URUN3	XYZ	900
U004	URUN3	ABC	550
U005	URUN2	ABC	850

ÖRNEK: Ürünler tablosu içerisinde markası ABC olan ve fiyatı 600 ün altında olan ürünlerin ürün kodlarını ve ürün isimlerini listeleyen ilişkisel cebir ifadesini yazınız.



# Atma (Projection) $\Pi$

ÜRÜNLER

<u>Urun_kod</u>	<u>Urun_adi</u>
U001	URUN1
U004	URUN3

$\Pi$ Urun\_kod, Urun\_adi( $\sigma$  Marka = ABC ^ Fiyat < 600(ÜRÜNLER))

# Kartezyen Çarpım (Cross-Product) X

- Belirli bir ilişkiden mümkün olabilecek tüm ilişki çiftlerinin elde edilmesi ve tek bir ilişki biçiminde gösterilmesi için kartezyen çarpım kullanılır.
- X sembolü ile gösterilmektedir.

# Kartezyen Çarpım (Cross-Product) X

TABLO1 X TABLO2

A
B
C

X
Y



A	X
A	Y
B	X
B	Y
C	X
C	Y

# Kartezyen Çarpım (Cross-Product) X

- Aşağıda verilen öğrenci ve dersler tabloları için kartezyen çarpımını uygulayınız.

Öğrenci

<u>Oğrenci no</u>	<u>Ogr adi</u>	<u>Bolum</u>
O001	Ahmet	Bilgisayar
O002	Ali	Elektrik
O003	Ela	Bilgisayar

Dersler

<u>Ders no</u>	<u>Ders adi</u>
BT102	VeriTabanı
BT103	Algoritma

# Kartezyen Çarpım (Cross-Product) X

<u>Ogrenci_no</u>	<u>Ogr_adi</u>	<u>Bolum</u>	<u>Ders_no</u>	<u>Ders_adi</u>
O001	Ahmet	Bilgisayar	BT102	Veritabanı
O001	Ahmet	Bilgisayar	BT103	Algoritma
O002	Ali	Elektrik	BT102	Veritabanı
O002	Ali	Elektrik	BT103	Algoritma
O003	Ela	Bilgisayar	BT102	Veritabanı
O003	Ela	Bilgisayar	BT103	Algoritma

Öğrenci X Dersler

# Kartezyen Çarpım (Cross-Product) X

- Aşağıdaki tabloları inceleyerek Ankaradaki depoda bulunan markaları ve tüm ürünleri listeleyen ilişkişel cebir ifadesini yazınız.

Ürünler

<u>Urun kod</u>	<u>Urun adi</u>	Marka	Fiyat
U001	URUN1	ABC	350
U002	URUN2	XYZ	750
U003	URUN3	XYZ	900
U004	URUN3	ABC	850
U005	URUN2	ABC	850

Markalar

<u>Marka</u>	<u>Üretici</u>	Depo
ABC	Üretici1	Ankara
XYZ	Üretici3	İzmir
KLM	Üretici1	Bursa
EDF	Üretici2	Ankara
DEF	Üretici1	İstanbul

# Kartezyen Çarpım (Cross-Product) X

- $\sigma$  Markalar.Depo = Ankara (ÜRÜNLER X MARKALAR)

<u>Urun kod</u>	<u>Urun adi</u>	Marka	Fiyat	Marka	Üretici	Depo
U001	URUN1	ABC	350	ABC	Üretici1	Ankara
U001	URUN1	ABC	350	EDF	Üretici2	Ankara
U002	URUN2	XYZ	750	ABC	Üretici1	Ankara
U002	URUN2	XYZ	750	EDF	Üretici2	Ankara
U003	URUN3	XYZ	900	ABC	Üretici1	Ankara
U003	URUN3	XYZ	900	EDF	Üretici2	Ankara
U004	URUN3	ABC	850	ABC	Üretici1	Ankara
U004	URUN3	ABC	850	EDF	Üretici2	Ankara
U005	URUN2	ABC	850	ABC	Üretici1	Ankara
U005	URUN2	ABC	850	EDF	Üretici2	Ankara



# Kartezyen Çarpım (Cross-Product) X

- İzmirde bulunan markalar ve tüm ürünler için ürün kodu, üretici ve ürün adını listeleyen ilişkişel cebir ifadesini yazınız.

Ürünler

<u>Urun_kod</u>	<u>Urun_adi</u>	Marka	Fiyat
U001	URUN1	ABC	350
U002	URUN2	XYZ	750
U003	URUN3	XYZ	900
U004	URUN3	ABC	850
U005	URUN2	ABC	850

Markalar

<u>Marka</u>	<u>Üretici</u>	Depo
ABC	Üretici1	Ankara
XYZ	Üretici3	İzmir
KLM	Üretici1	Bursa
EDF	Üretici2	Ankara
DEF	Üretici1	İstanbul

# Kartezyen Çarpım (Cross-Product) X

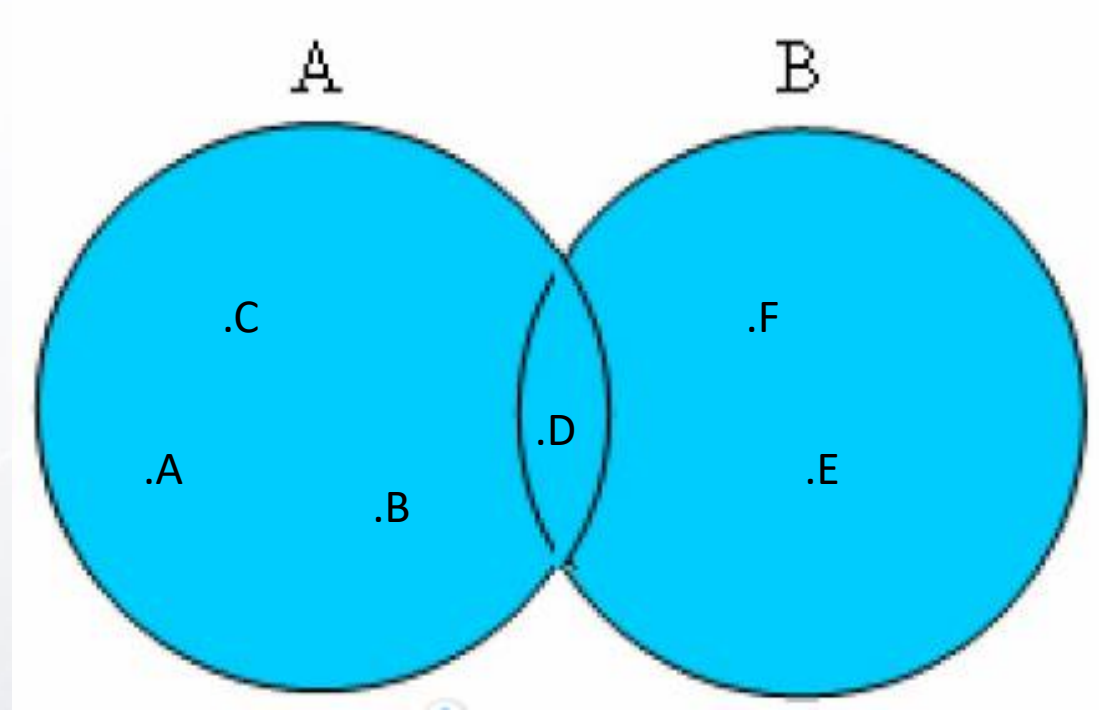
$\Pi$ Urun\_kod, Urun\_adi, Üretici( $\sigma$  Markalar.depo = İzmir(ÜRÜNLER X MARKALAR))

<u>Ürün kod</u>	<u>Ürün adi</u>	Üretici
U001	URUN1	Üretici3
U002	URUN2	Üretici3
U003	URUN3	Üretici3
U004	URUN3	Üretici3
U005	URUN2	Üretici3

# Birleşim (Union) U

- Kurulan iki ilişkiden birinde veya her ikisinde birden bulunan kayıtların seçilmesi için yapılan bir işlem türüdür.
- U ile gösterilir.

Tablo1 U Tablo2



# Birleşim (Union) U

- Bütün bu işlemler girdi olarak iki ilişki alır, ki bu iki ilişki birleşime uyumlu olmalıdır, bu da demek oluyor ki:
- Aynı sayıda alana sahip olmalı.
- Karşılıklı alanlar aynı tipte olmalı.

# Birleşim (Union) U

Tablo1

<u>sira_no</u>	<u>adi</u>	Soyadi
1	AHMET	DEMİR
2	BÜŞRA	UÇAR
3	FUAT	CEBE

Tablo2

<u>sira_no</u>	<u>adi</u>	Soyadi
1	MURAT	BARUT
2	AYŞE	KAYA

Yukarıdaki tabloları birleştirmek için gerekli ilişkisel cebir ifadesi nedir?

# Birleşim (Union) U

## Tablo1 U Tablo2

<u>sira_no</u>	<u>adi</u>	Soyadi
1	AHMET	DEMİR
2	BÜŞRA	UÇAR
3	FUAT	CEBE
1	MURAT	BARUT
2	AYŞE	KAYA

# Birleşim (Union) U

Tablo1

<u>sira_no</u>	<u>adi</u>	Soyadi
1	AHMET	DEMİR
2	AYŞE	KAYA
3	FUAT	CEBE

Tablo2

<u>sira_no</u>	<u>adi</u>	Soyadi
1	MURAT	BARUT
2	AYŞE	KAYA

Yukarıdaki tabloların birleşiminden sadece ad bilgilerinin listeleyecek ilişkisel cebir ifadesini yazınız.

# Birleşim (Union) U

$\Pi$  adi(Tablo1 U Tablo2)

<u>adi</u>
AHMET
AYŞE
FUAT
MURAT

NOT: iki farklı tabloda bulunan aynı kayıtların sadece bir tanesi listelenir.



# Birleşim (Union) U

Depo1

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250
ÜRÜN2	500
ÜRÜN3	300

Depo2

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250
ÜRÜN2	700

Depo1 de fiyatı 400 ve üzeri olan ürünlerle depo2'de bulunan ve fiyatı 500 ve üzerinde olan ürünlerin adlarını listeleyen ilişkisel cebir ifadesini yazınız.

# Birleşim (Union) U

$\Pi$ ürün\_adi

$(\sigma \text{ fiyat} \geq 400(\text{DEPO1}))$

U

$(\sigma \text{ fiyat} \geq 500(\text{DEPO2}))$

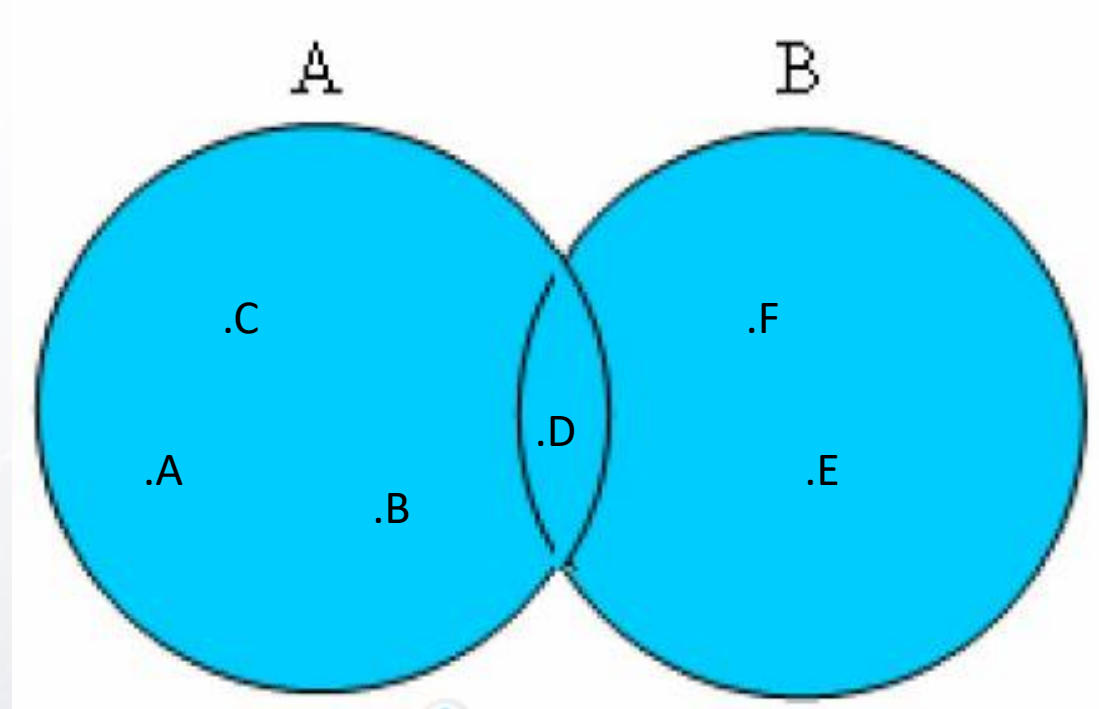
<u>Ürün adı</u>
ÜRÜN2
ÜRÜN4

# Küme Farkı (Set-difference) -

- İki ilişkiden birisinde bulunup diğerinde bulunmayan satırları bulmak için kullanılır.
- Belirtilen iki ilişki de aynı sayıda sütuna sahip olmalıdır ve karşılıklı sütunlar aynı tipte olmalıdır.
- - ile gösterilir.

# Küme Farkı (Set-difference) -

- $A - B : A, B, C$
- $B - A : F, E$



# Küme Farkı (Set-difference) -

Depo1

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250
ÜRÜN2	500
ÜRÜN3	300

Depo2

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250
ÜRÜN4	700

Depo1 – Depo2

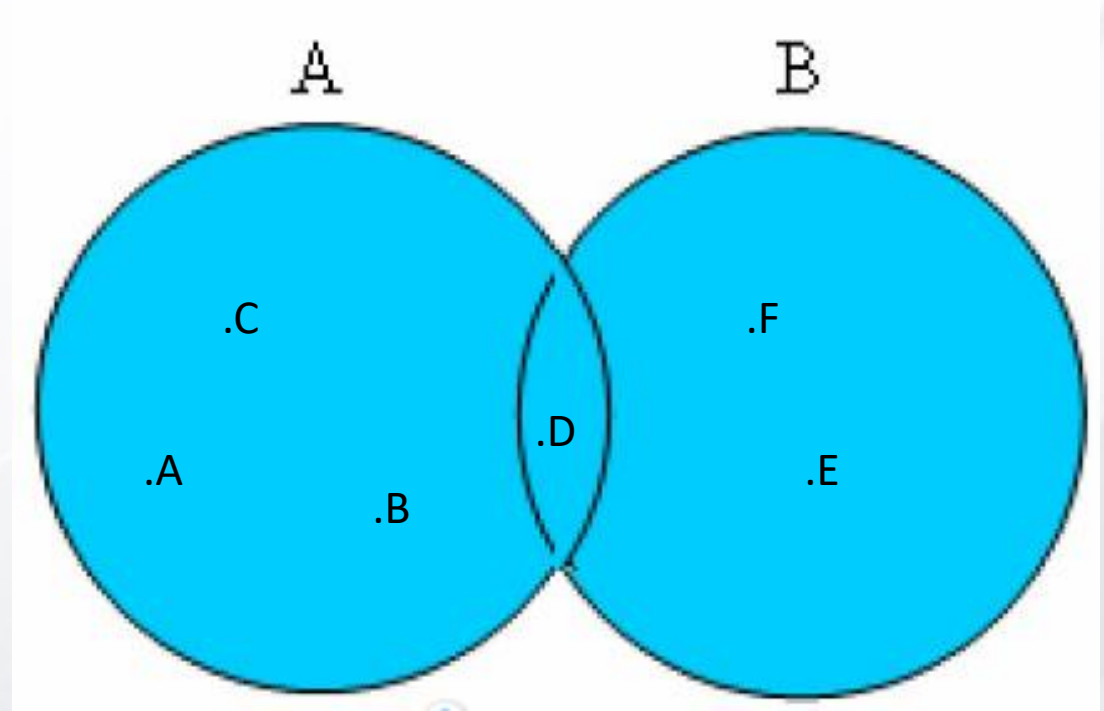
<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN2	500
ÜRÜN3	300

# Kesişim (Intersection)

- İki ilişkiden her ikisinde de bulunan satırları almak için kullanılır.
- Belirtilen iki ilişki de aynı sayıda sütuna sahip olmalıdır ve karşılıklı sütunlar aynı tipte olmalıdır.
- $\cap$  ile gösterilir.

# Kesişim (Intersection)

•  $A \cap B : D$



# Kesişim (Intersection)

Depo1

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250
ÜRÜN2	500
ÜRÜN3	300

Depo2

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250
ÜRÜN4	700

Depo1  $\cap$  Depo2

<u>Ürün adı</u>	<u>Fiyatı</u>
ÜRÜN1	250



# Bölme (Division) :

- Elde edilen iki ilişkiyi karşılaştırarak, birinci ilişkide ikinci ilişkinin tüm elemanlarını kapsayan satırları bulmak için kullanılır.
- Tablo1 : Tablo2

# Bölme (Division) :

Sıra_no	Ad_soyad	Yaş
1	Ahmet Demir	25
2	Büşra Uçar	25
3	Aydın Cebe	45

Yaş
25

## Personel : Yaş

Sıra_no	Ad_soyad
1	Ahmet Demir
2	Büşra Uçar

# Bölme (Division) :

SATIŞLAR

<u>Sno</u>	Cari	Ürün
1	Cari1	Ürün1
2	Cari2	Ürün2
3	Cari1	Ürün3
4	Cari2	Ürün1
5	Cari1	Ürün2

ÜRÜNLER

<u>Sno</u>	Ürün	Fiyat	Depo
1	Ürün1	350	Ankara
2	Ürün2	600	Ankara
3	Ürün3	250	İzmir

- Yukarıdaki tablolar için tüm ürünlerden de satın alan carileri listeleyen ilişkisel cebir ifadesini yazınız.

# Bölme (Division) :

- Ürünler tablosundan ürün isimlerini listele:

$$K = \prod_{\text{ürünler.ürün}}(\text{ÜRÜNLER})$$

ÜRÜNLER

Ürün

Ürün1

Ürün2

Ürün3

# Bölme (Division) :

- Satışlar tablosundan cari ve ürün sütunlarını alalım

$$L = \prod \text{satışlar.cari, satışlar.ürün}(\text{SATIŞLAR})$$

SATIŞLAR

Cari	Ürün
Cari1	Ürün1
Cari2	Ürün2
Cari1	Ürün3
Cari2	Ürün1
Cari1	Ürün2

# Bölme (Division) :

- L:K işlemini uygulayalım:

**⌈satışlar.cari, satışlar.ürün(SATIŞLAR) : ⌈ürünler.ürün(ÜRÜNLER)**

Cari

Cari1

# Birleştirme (join)

- ŞARTLI BİRLEŞTİRME (CONDITION JOIN)
- EŞİT BİRLEŞTİRME (EQUIJOIN)
- DOĞAL BİRLEŞTİRME (NATURAL JOIN)
- DIŞSAL BİRLEŞTİRME (OUTER JOIN)
  - SOL BİRLEŞTİRME (LEFT OUTER JOIN)
  - SAĞ BİRLEŞTİRME (RIGHT OUTER JOIN)
  - TAM BİRLEŞTİRME (FULL OUTER JOIN)
- YARI BİRLEŞTİRME (SEMI-JOIN)
- ANTİ JOIN

# Birleřtirme (join)

## **DOĐAL BİRLEŐTİRME (Natural Join)**

Birleřtirilen iliřkiler arasında herhangi bir kořul ifadesi yazmaya gerek yoktur. Ortak sřtunlar otomatik birleřtirilir ve tek bir sřtun olarak gřsterilir.



# Birleştirme (join)

Tablo1 ⋈ Tablo2

**Örnek:** Aşağıda öğrenci ve bölüm tablosu verilmiştir. Öğrenci tablosunda bölüm bilgisi bölüm kodu şeklinde ifade edilmektedir. Bölüm kodu ise bölüm tablosunun birinci anahtarıdır ve bölüm adlarını da içermektedir. Bu bilgiler doğrultusunda öğrenci no, isim ve bölüm isimlerini göstermek için aşağıdaki ilişkisel cebir ifadesi kullanılır.

Ogr_no	Ogr_adi	Bolum_kod
2010001	Turgut Özseven	BTP
2010002	Ahmet Kaçar	ELK
2010002	Mustafa Yıldız	ELKT
2010003	Elif Kara	BTP
2010004	Mustafa Çağlayan	MKN

Bolum_kod	Bölüm
BTP	Bilgisayar
ELK	Elektrik
ELKT	Elektronik
MKN	Makine

# Birleştirme (join)

Öğrenci ⋈ Bölüm

Ogr_no	Ogr_adi	Bolum_kod	Bölüm
2010001	Turgut Özseven	BTP	Bilgisayar
2010002	Ahmet Kaçar	ELK	Elektrik
2010002	Mustafa Yıldız	ELKT	Elektronik
2010003	Elif Kara	BTP	Bilgisayar
2010004	Mustafa Çağlayan	MKN	Makine

# Birleştirme (join)

## EŞİT BİRLEŞTİRME (Equijoin)

Birleştirilen ilişkiler arasındaki ortak sütunlar için eşitlik ifadesi belirtilir. Elde edilen sonuç ilişkisi Kartezyen çarpıma benzer ama ortak sütunlardan sadece bir tanesi vardır.

Tablo1  $\bowtie_e$  Tablo2

# Birleştirme (join)

**Örnek:** Aşağıda ürünlerin alım ve satış fiyatlarını içeren tablolar verilmiştir. Ürünlerin alım ve satıl fiyatlarını göstermek için aşağıdaki ilişkisel cebir ifadesi kullanılır.

SATIŞ		
Sno	Ürün	Fiyat
1	Ürün1	500
2	Ürün2	300
3	Ürün3	700

ALIM		
Sno	Ürün	Fiyat
1	Ürün1	350
2	Ürün2	350
3	Ürün3	400

Satış  $\bowtie$  alım.ürün=satış.ürün Alım

Sno	Ürün	Fiyat	Sno	Fiyat
1	Ürün1	350	1	500
2	Ürün2	350	2	300
3	Ürün3	400	3	700

# Birleřtirme (join)

## **ŐARTLI BİRLEŐTİRME (Condition Join)**

Birleřtirilen iliřkiler arasındaki ortak sütünlar için kořul ifadesi belirtilir.

# Şartlı Birleştirme

Alım fiyatı satış fiyatından yüksek ürünleri bulan ilişkisel cebir ifadesi

SATIŞ		
Sno	Ürün	Fiyat
1	Ürün1	500
2	Ürün2	300
3	Ürün3	700

ALIM		
Sno	Ürün	Fiyat
1	Ürün1	350
2	Ürün2	350
3	Ürün3	400

Satış  $\bowtie$   $\text{alim.fiyat} > \text{satış.fiyat}$  Alım

Sno	Ürün	Fiyat	Sno	Ürün	Fiyat
1	Ürün1	350	2	Ürün2	300
2	Ürün2	350	2	Ürün2	300
3	Ürün3	400	2	Ürün2	300

# Şartlı Birleştirme

Alım fiyatı satış fiyatından yüksek ürünleri bulan ilişkisel cebir ifadesi

SATIŞ		
Sno	Ürün	Fiyat
1	Ürün1	500
2	Ürün2	300
3	Ürün3	700

ALIM		
Sno	Ürün	Fiyat
1	Ürün1	350
2	Ürün2	350
3	Ürün3	400

Satış  $\bowtie$   $\text{alim.fiyat} > \text{satış.fiyat}$  Alım

Sno	Ürün	Fiyat	Sno	Ürün	Fiyat
1	Ürün1	350	2	Ürün2	300
2	Ürün2	350	2	Ürün2	300
3	Ürün3	400	2	Ürün2	300

# Birleřtirme (join)

## DIŐSAL BİRLEŐTİRMELER (OUTER JOIN)

Normal join iřleminde iliřkili olmayan satırlar gosterilmemektedir. Outer join de ise iliřkili olmayan satırlar da gosterilecektir ama iliřkisi olmayan satırlar NULL deęer ięerecektir.



# Outer Join

## LEFT OUTER JOIN

Birleştirme sembolünün solundaki ilişki belirleyicidir ve bunun diğer ilişkiyle bir ilişkisi bulunsun veya bulunmasın tüm satırları listelenecektir.

# Left Outer Join

Tablo1  $\bowtie$  Tablo2

**Örnek:** Aşağıda verilen öğrenci ve bölüm tablosunu left outer join ile birleştirelim.

Ogr_no	Ogr_adi	Bolum_kod
2010001	Turgut Özseven	BTP
2010002	Ahmet Kaçar	ELK
2010002	Mustafa Yıldız	ELKT
2010003	Elif Kara	TBD
2010004	Mustafa Çağlayan	MKN

Bolum_kod	Bölüm
BTP	Bilgisayar
ELK	Elektrik
ELKT	Elektronik
MKN	Makine

$\Pi_{ogr\_no,ogr\_adi,bölüm}(\text{Öğrenci} \bowtie_{\text{öğrenci.bolum\_kod=bölüm.bölüm\_kod}} \text{Bölüm})$

# Left Outer Join

$\Pi_{ogr\_no,ogr\_adi,bölüm}(\text{Öğrenci} \bowtie_{\text{öğrenci.bolum\_kod=bölüm.bölüm\_kod}} \text{Bölüm})$

Ogr_no	Ogr_adi	Bölüm
2010001	Turgut Özseven	Bilgisayar
2010002	Ahmet Kaçar	Elektrik
2010002	Mustafa Yıldız	Elektronik
2010003	Elif Kara	NULL
2010004	Mustafa Çağlayan	Makine

# Birleřtirme (join)

## RIGHT OUTER JOIN

Birleřtirme sembolünün sađındaki iliřki belirleyicidir ve bunun diđer iliřkiyle bir iliřkisi bulunsun veya bulunmasın tüm satırları listelenecektir.

# Right Outer Join

Right outer join  $\bowtie$  sembolü ile gösterilir ve aşağıdaki şekilde kullanılır.

Tablo1  $\bowtie$  Tablo2

**Örnek:** Aşağıdaki öğrenci ve bölüm tablosunu birleştirerek tüm bölümleri ve öğrencilerin bölümlerinin isimlerini yazdıralım

Ogr_no	Ogr_adi	Bolum_kod
09001	Ahmet Demir	1
09002	Büşra Uçar	5
09003	Fuat Cebe	3

Bolum_kod	Bölüm
1	Bilgisayar
2	Elektrik
3	Tesisat

$\Pi_{ogr\_no,ogr\_adi,bölüm}(\text{Öğrenci} \bowtie_{\text{öğrenci.bolum\_kod=bölüm.bölüm\_kod}} \text{Bölüm})$

# Right Outer Join

ogr_no	adsoyad	bolum_adi
09001	Ahmet Demir	Bilgisayar
NULL	NULL	Elektrik
09003	Fuat Cebe	Tesisat

# Birleştirme (join)

## FULL OUTER JOIN

Left outer join ve right outer join işlemlerinin birleşim kümesidir.

Her iki ilişkideki tüm satırlar listelenir ve ilişkisi olmayan satırlar NULL ile doldurulur.

# Full Outer Join

Tablo1  $\bowtie$  Tablo2

**Örnek:** Aşağıdaki öğrenci ve bölüm tablosunu birleştirerek tüm bölümleri ve tüm öğrencilerin bölümlerinin isimlerini yazdıralım

Ogr_no	Ogr_adi	Bolum_kod
09001	Ahmet Demir	1
09002	Büşra Uçar	5
09003	Fuat Cebe	3

Bolum_kod	Bölüm
1	Bilgisayar
2	Elektrik
3	Tesisat

$\Pi_{ogr\_no,ogr\_adi,bölüm}(\text{Öğrenci} \bowtie_{\text{öğrenci.bolum\_kod=bölüm.bölüm\_kod}} \text{Bölüm})$



# Full Outer Join

$\Pi_{ogr\_no,ogr\_adi,bölüm}(\text{Öğrenci} \bowtie_{ogr\_no.bolum\_kod=bölüm.bölüm\_kod} \text{Bölüm})$

ogr_no	adsoyad	bolum_adi
09001	Ahmet Demir	Bilgisayar
09002	Büşra Uçar	NULL
09003	Fuat Cebe	Tesisat
NULL	NULL	Elektrik

# Yarı Birleştirme (Semi-Join)

Semi-Join doğal birleştirmeye benzer şekilde çalışır. Farklı olarak sadece belirtilen ilişkilerden bir tanesinin sütun bilgilerini içerir, diğer ilişkinin sütun bilgilerini içermez.

Semi-join  $\bowtie$  sembolü ile gösterilir ve aşağıdaki şekilde kullanılır.

Tablo1  $\bowtie$  Tablo2

# Yarı Birleştirme (Semi-Join)

**Örnek:** Aşağıdaki öğrenci ve bölüm tablosu için sadece bölüm tablosunda kayıtlı bölümleri olan öğrencileri seçmek için aşağıdaki ilişkisel cebir ifadesi kullanılır.

Ogr_no	Ogr_adi	Bolum_kod
09001	Ahmet Demir	1
09002	Büşra Uçar	5
09003	Fuat Cebe	3

Bolum_kod	Bölüm
1	Bilgisayar
2	Elektrik
3	Tesisat

# Anti-Join

Anti-join, semi-join gibi ilişkilerden birisinde bulunan sütunları gösterir ama işlev olarak tam tersidir. İki ilişki arasında bağ bulunmayan satırları belirlemek için kullanılır.

Anti-join  $\triangleright$  sembolü ile gösterilir ve aşağıdaki şekilde kullanılır.

Tablo1  $\triangleright$  Tablo2

# Anti-Join

**Örnek:** Aşağıdaki öğrenci ve bölüm tablosu için bölüm kodu bilgisi bölümler tablosunda bulunmayan öğrencileri bulmak için aşağıdaki ilişkisel cebir ifadesi kullanılır.

Ogr_no	Ogr_adi	Bolum_kod
09001	Ahmet Demir	1
09002	Büşra Uçar	5
09003	Fuat Cebe	3

Bolum_kod	Bölüm
1	Bilgisayar
2	Elektrik
3	Tesisat

Öğrenci  $\triangleright$  öğrenci.bolum\_kod=bölüm.bölüm\_kod Bölüm

ogr_no	adsoyad	Bolum_kod
09002	Büşra Uçar	5

# Yeniden Adlandırma

- İlişkiler veya ilişkilerin içerdiği sütunlara takma isim vermek için kullanılır.
- Birleştirme veya Kartezyen çarpım sonucu elde edilen aynı isimli sütunların karışıklığını önlemek veya ilişkisel cebir sonucu oluşan tablolara isim vermek için kullanılabilir.
- P ile gösterilir.

# Yeniden Adlandırma

- Kullanımı

- $P_s(b_1, b_2, \dots, b_n)(R)$

Sütunların Takma İsimleri

İlişkinin Takma İsmi

- $P_s(R)$

- $P(b_1, b_2, \dots, b_n)(R)$

# Yeniden Adlandırma

Ogr_no	Ogr_adi	Bolum_kod
09001	Ahmet Demir	1
09002	Büşra Uçar	5
09003	Fuat Cebe	3

Bolum_kod	Bölüm
1	Bilgisayar
2	Elektrik
3	Tesisat

- Yukarıda verilmiş olan tabloları birleştirerek oluşan yeni tablonun sütun isimlerini değiştirin.
- $(P(\text{no}, \text{ad}, \text{bkod})(\text{Öğrenci})) \bowtie \text{bkod} = \text{bkod2}(P(\text{bkod2}, \text{bad})(\text{Bolum}))$



# Yeniden Adlandırma

No	Ad	Bkod	Bad
09001	Ahmet Demir	1	Bilgisayar
09003	Fuat Cebe	3	Tesisat

# Yeniden Adlandırma

CALIŞANLAR

Sno	İsim	Yönetici
1	Turgut	NULL
2	Ahmet	3
3	Murat	1
4	Ayşe	3
5	Elif	1

Örnek 1: Ayşe isimli personelin yöneticisinin ismini bulan ilişkisel cebir İfadesini yazınız.

$$\prod_{yisim} \left( \varphi_{\text{çisim}=Ayşe} P_{\text{çalışan}}(\text{çno}, \text{çisim}, \text{çyon})(\text{Çalışanlar}) \mid x \mid \text{çyon}=\text{yno} \right. \\ \left. P_{\text{yonetici}}(\text{yno}, yisim, yyon)(\text{Çalışanlar}) \right)$$

# Yeniden Adlandırma

Sno	İsim	Yönetici
1	Turgut	NULL
2	Ahmet	3
3	Murat	1
4	Ayşe	3
5	Elif	1

Örnek 2: Ayşe isimli personelin yöneticisinin yöneticisini isim olarak bulan ilişkisel cebir ifadesini yazınız.

# Yeniden Adlandırma

$\prod_{yisim} (\varphi_{\text{çisim=Ayşe}} (P_{\text{çalışan}}(\text{çno}, \text{çisim}, \text{çyon})(\text{Çalışanlar}) \mid x \mid \text{çyon}=\text{yno}$   
 $P_{\text{yonetici}}(\text{yno}, \text{yisim}, \text{yyon})(\text{Çalışanlar}))$   
 $\mid x \mid \text{yyon}=\text{ustno} (P_{\text{ustyonetici}}(\text{ustno},$   
 $\text{ustisim}, \text{ustyno})(\text{Çalışanlar})))$

# Özetleme ve Gruplama

- Özetleme işlemi bir tablo içeriğinin istenilen bir sütuna göre hesaplamalar yapılmasını içerir.
- Gruplama ise bir tablonun içeriğini istenilen bir sütun içeriğine göre gruplandırarak her bir grup için hesaplamalar yapılmasını sağlar.
- Kullanılan fonksiyonlar:  
SUM, COUNT, AVERAGE, MAX, MIN

# Özetleme ve Gruplama

Kullanım şekli aşağıda verilmiştir.

[Gruplandırılacak sütun]  $\Sigma$  [fonksiyon adı][sütun adı]

[Gruplandırılacak sütun] parametresi, özetleme işlemi gruplandırarak yapılacaksa kullanılır. Sadece özetleme işlemi için bu parametrenin kullanımına gerek yoktur.

[fonksiyon adı] yapılan gruplandırma işlemi için kullanılacak olan fonksiyonu belirler(SUM, COUNT, AVERAGE, MAX ve MIN).

[sütun adı] belirlenen fonksiyonun hangi sütun için uygulanacağını belirler.

# Özetleme ve Gruplama

Sno	İsim	Yönetici	Maas
1	Turgut	NULL	1500
2	Ahmet	3	1000
3	Murat	1	1100
4	Ayşe	3	900
5	Elif	1	1000

1. Tüm personellerin sayısı  $t_{\text{COUNT}}(\text{Sno})$
2. Personellerin ortalama maaş tutarını  $t_{\text{AVARAGE}}(\text{Maas})$
3. En yüksek ve en düşük maaşa sahip çalışanları bulunuz.  
 $t_{\text{MAX}}(\text{Maas})$   $t_{\text{MIN}}(\text{Maas})$

# Özetleme ve Gruplama

Sno	İsim	Yönetici	Maas
1	Turgut	NULL	1500
2	Ahmet	3	1000
3	Murat	1	1100
4	Ayşe	3	900
5	Elif	1	1000

1. Yönetici olan personellerin yönetimi altında çalışan personel sayısını,
2. Maaşı 1000 in üzerinde olan personellerin sayısını bulunuz.

1) yönetici  $t_{COUNT\ sno}$  (Çalışanlar)

2)  $t_{count\ sno}$  ( $\Phi_{Maas > 1000}$ ) (Çalışanlar)



Opel Astra ve Renault Megane marka araçların her ikisinden de kiralayan müşterilerin ad, soyad ve telefon numarası bilgilerini bulunuz.



Araç Tablosu				
Aracno	Model	Marka	Plaka	Fiyat
A001	2004	Fiat Marea	60 TT 6060	10
A002	2000	Renault Megane	60 TT 6061	7
A003	2007	Ford Focus	60 TT 6062	15
A004	2005	Volkswagen Golf	60 TT 6063	14
A005	1998	Opel Astra	60 TT 6064	6

Müşteri Tablosu				
Mkod	Mad	Msoyad	Madres	Mtel
M001	Turgut	Özseven	Turhal/Tokat	03562222222
M002	Mustafa	Çağlayan	Meram/Konya	05112111111
M003	Ahmet	Kara	Zile/Tokat	03563333333
M004	Murat	Beyaz	Turhal/Tokat	03565555555
M005	Elif	Kurt	Beşiktaş/İstanbul	05781471414
M006	Ayşe	Uçar	Taşova/Amasya	03586666666
M007	Bülent	Ayar	Turhal/Tokat	03568888888

Kiralama Tablosu					
Mkod	Aracno	Tarih	Saat	Tes_tarihi	Tes_saat
M001	A004	01.05.2010	14:00	03.05.2010	14:00
M003	A001	04.05.2010	12:00	05.05.2010	14:00
M001	A003	01.06.2010	14:00	04.06.2010	17:00
M002	A005	10.06.2010	10:00	12.06.2010	15:00
M004	A002	10.06.2010	12:00	11.06.2010	16:00
M001	A001	11.06.2010	09:00	15.06.2010	13:00
M003	A004	15.06.2010	15:00	16.06.2010	18:00
M005	A005	02.07.2010	11:30	04.07.2010	12:00
M002	A002	05.07.2010	13:30	08.07.2010	15:00
M003	A003	15.07.2010	14:00	20.07.2010	15:00

# İlişkisel Cebir Örnekleri



# İlişkisel Cebir Örnekleri

<b>Araç Tablosu</b>				
<b>Aracno</b>	<b>Model</b>	<b>Marka</b>	<b>Plaka</b>	<b>Fiyat</b>
A001	2004	Fiat Marea	60 TT 6060	10
A002	2000	Renault Megane	60 TT 6061	7
A003	2007	Ford Focus	60 TT 6062	15
A004	2005	Volkswagen Golf	60 TT 6063	14
A005	1998	Opel Astra	60 TT 6064	6

# İlişkisel Cebir Örnekleri

<b>Müşteri Tablosu</b>				
<b>Mkod</b>	<b>Mad</b>	<b>Msoyad</b>	<b>Madres</b>	<b>Mtel</b>
M001	Turgut	Özseven	Turhal/Tokat	03562222222
M002	Mustafa	Çağlayan	Meram/Konya	05112111111
M003	Ahmet	Kara	Zile/Tokat	03563333333
M004	Murat	Beyaz	Turhal/Tokat	03565555555
M005	Elif	Kurt	Beşiktaş/İstanbul	05781471414
M006	Ayşe	Uçar	Taşova/Amasya	03586666666
M007	Bülent	Ayar	Turhal/Tokat	03568888888

# İlişkisel Cebir Örnekleri

Kiralama Tablosu					
Mkod	Aracno	Tarih	Saat	Tes_tarihi	Tes_saat
M001	A004	01.05.2010	14:00	03.05.2010	14:00
M003	A001	04.05.2010	12:00	05.05.2010	14:00
M001	A003	01.06.2010	14:00	04.06.2010	17:00
M002	A005	10.06.2010	10:00	12.06.2010	15:00
M004	A002	10.06.2010	12:00	11.06.2010	16:00
M001	A001	11.06.2010	09:00	15.06.2010	13:00
M003	A004	15.06.2010	15:00	16.06.2010	18:00
M005	A005	02.07.2010	11:30	04.07.2010	12:00
M002	A002	05.07.2010	13:30	08.07.2010	15:00
M003	A003	15.07.2010	14:00	20.07.2010	15:00

# İlişkisel Cebir Örnekleri

- Soru1: A004 kodlu aracı kiralayan müşterilerin ad, soyad bilgilerini bulunuz.
- Soru2: Fiat Marea marka aracı kiralayan müşterilerin ad, soyad bilgilerini bulunuz.
- Soru3: Mustafa Çağlayan isimli müşterinin kiraladığı araç veya araçların markasını bulunuz.
- Soru 4: Opel Astra ve Renault Megane marka araçların her ikisinden de kiralayan müşterilerin ad, soyad ve telefon numarası bilgilerini bulunuz.

Soru1: A004 kodlu aracı kiralayan müşterilerin ad, soyad bilgilerini bulunuz.

$\Pi_{\text{Mad, Msoyad}} (\varphi_{\text{AraçKod=A004(Kiralama)} | X | \text{MÜŞTERİ})$

Müşteri Tablosu				
Mkod	Mad	Msoyad	Madres	Mtel

Mkod	Aracno	Tarih	Saat	Tes_tarihi	Tes_saat
M001	A004	01.05.2010	14:00	03.05.2010	14:00
M003	A001	04.05.2010	12:00	05.05.2010	14:00

Sorgu2 Fiat Marea marka aracı kiralayan müşterilerin ad, soyad bilgilerini bulunuz.

$\prod_{\text{Mad, Msoyad}} (\varphi_{\text{Marka=Fiat Marea(Araçlar) |X| KIRALAMA}}) |X|$   
MÜŞTERİ

Aracno	Model	Marka	Plaka	Fiyat	Mkod	Mad	Msoyad	Madres	Mtel
--------	-------	-------	-------	-------	------	-----	--------	--------	------

Mkod	Aracno	Tarih	Saat	Tes_tarihi	Tes_saat
------	--------	-------	------	------------	----------



Sorgu 3 Mustafa Çağlayan isimli müşterinin kiraladığı araç veya araçların markasını bulunuz.

$$\Pi_{\text{MARKA}} \left( \left( \varphi_{\text{ad=Mustafa} \wedge \text{soyad= Çağlayan (Müşteriler)} \mid X \mid \text{KİRALAMA} \right) \mid X \mid \text{ARAÇLAR} \right)$$

Sorgu 4 Opel Astra ve Renault Megane marka araçların her ikisinden de kiralayan müşterilerin ad, soyad ve telefon numarası bilgilerini bulunuz.

$P_{\text{opelkira}} \prod_{\text{MKOD}} ( \varphi_{\text{Marka=OpelAstra (Araçlar) |X| KİRALAMA}} )$

$P_{\text{meganekira}} \prod_{\text{MKOD}} ( \varphi_{\text{Marka=Megane (Araçlar) |X| KİRALAMA}} )$

$P_{\text{musterikod}}(\text{Opelkira} \cap \text{meganekira})$

$\prod_{\text{Mad, Msoyad, Mtel}} (\text{Musterikod} \langle \text{join} \rangle \text{MÜSTERİLER})$

<input type="text" value="Mkod"/>	<input type="text" value="Mkod"/>	<input type="text" value="Mad"/>	<input type="text" value="Msoyad"/>	<input type="text" value="Madres"/>	<input type="text" value="Mtel"/>
-----------------------------------	-----------------------------------	----------------------------------	-------------------------------------	-------------------------------------	-----------------------------------