

Veritabanı Yönetim Sistemleri (335)

Yrd. Doç. Dr. Ahmet Arif AYDIN

Agenda

- ilişkisel veri modeli (relational data model)
- ilişkisel Cebir (relational algebra)

Relational Data Model

İlişkisel veri modeli (relational data model)

- 1970 yılında IBM'in araştırmacılarından Edgar F. Codd tarafından geliştirilmiştir.
- Günümüzde kullanılan bir çok VTYS ilişkisel veri modelini(kullanmaktadır.
 - IBM's DB2 , Informix
 - Oracle, Sybase
 - Microsoft's Access , FoxBase
 - Paradox, Tandem

Relational Data Model

Verinin tablolarda satırlar ve sütunlar olarak saklandığı
tablolar arasında ilişkilerin tanımlanabildiği,
verinin tutarlılığının sağlanabildiği sistem
ilişkisel veritabanı yönetim sistemleri olarak tanımlanmaktadır

Relational Data Model

- Bir veritabanı bir veya daha fazla ilişkinin oluşturduğu koleksiyondur.
- Her bir ilişki ise satır ve sütunlardan oluşan tablolardır.

Relational Data Model: Avantajlar

Hiyerarsik ve ağ modellerinde bulunmayan ve ilişkisel veri modelinin sağladığı avantajlar:

- tablolar yardımıyla veriler veritabanında kaydedilir
- verinin yapısı daha kolay bir biçimde anlaşılır
- tablo yapısı ile birlikte veri üzerinde yapılacak sorgulamalar kolaylaştırılır

Relational Data Model: Şema

İlişkisel modelde veri tabloları kaydedilmekte ve şema (schema) olarak tanımlanmaktadır.

öğrenci (öğrencino: integer, isim: string, kullanıcıadı: string, yaş: integer, ortalama:real)

öğrenci tablosunun şeması



şemaya yardımıyla

- tablonun adı "öğrenci"
- her bir satır 5 sütundan oluşmaktadır.
- her bir sütunda kaydedilecek verinin tipini belirlemektedir

Relational Data Model: Tablo

kolonlar (nitelikler, alan)

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

satır (kayıt, tuple)

- Her bir satır bir öğrencinin bilgilerini içermektedir
- Her satır unique olmak zorundadır
- Her satır aynı şemayı izlemek zorundadır.
- Öğrenci tablosunda 5 satır bulunmakta ve her bir satır da 5 sütundan oluşmaktadır.
- İlişkisel model geleneğinde her bir satırda bulunan alanlar **aynı sırada** olmalıdır.

Relational Data Model: Tablo

kolonlar (nitelikler, alan)

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

satır (kayıt, tuple)

- Bu tablonun derecesi (degree of relation) kolon sayıdır 5 ve bu örneğin eleman sayısı (cardinality) 5 dir.
- Şema oluşturulduktan sonra yeni alan eklenemez
- Kaydedilen verinin tipi degistirilemez.
- Veritabanında bulunan tablo isimleri de farklı olmak zorundadır.

Relational Data Model: Tabloların Bütünlük Kısıtlamaları

- Bir veritabanı verilerin bütünlüğünü ve doğruluğunu korumak zorundadır.
- Bütünlük kısıtlamalarını tabloların tasarımı yapılırken ve oluşturulurken tanımlanan şartlardır.
- VTYS oluşturulan tabloların kısıtlamalarına uyan verileri tablolarda saklamaktadır.

Relational Data Model: Tabloların Bütünlük Kısıtlamaları

Öğrenci tablosunun oluşturulmasında ki bazı kısıtlamalar şunlardır:

- isim alanı rakamlardan oluşamaz
- yaş sütununda integer tipinde rakam bulunması gerekmektedir.
- Her bir öğrencinin numası diğerlerinden farklı (unique) olmak zorundadır.

kolonlar (nitelikler, alan)

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

satır (kayıt, tuple)

öğrenci (öğrencino: integer, isim: string, kullanıcıadı: string, yaş: integer, ortalama:real)

Relational Data Model: Tabloların Bütünlük Kısıtlamaları

- Bir tabloda kaydedilecek olan verileri diğerlerinden ayırt etmemizi sağlayan nitelik anahtar (key) olarak tanımlanmaktadır.
- Anahtar seçimi veritabanının bütünlüğü ve istenilen verinin sağlıklı bir biçimde kaydedilmesi açısından çok önemlidir.

kolonlar (nitelikler, alan)

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

satır (kayıt, tuple)

Relational Data Model: Primary Key

Öğrenci Tablosu

primary key

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

- Veritabanı tasarımında birincil anahtar (primary key) seçilmesi gerekmektedir.
- Öğrenci tablosunun öğrencino alanı primark key olarak tanımlanabilir
- Bu seçim ile birlikte isim benzerliği olan kayıtların doğru bir biçimde kaydedilmesi sağlanır.

Relational Data Model: Primary Key

Öğrenci Tablosu

primary key

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

Primary key de bulunması gereken özellikler:

- **Null** değer içermemelidir
- Kayıt işleminden sonra değişmemelidir
- Boyutu mümkün olduğu kadar az olmalıdır
- Şifrelenmiş değer içermemelidir.

Relational Data Model: foreign Key

- Bir tablodaki alanın başka bir tablodaki alanla ilişkilendirilmesinde yabancı anahtar (foreign key) kullanılmaktadır.
- Dersler tablosunun öğrencino alanı Öğrenci tablosunun öğrencino alanı ile ilişkilendirilip foreign key olarak tanımlanabilir.
- Foreign key olarak tanımlanan alandan birden fazla olabilir.

Dersler Tablosu

foreign key

öğrencino	derskodu	kredi	notu
17532	BL44	4	85
17327	MK25	3	73
17532	BL13	3	94
17236	T121	2	45
16458	KM32	1	52

Relational Data Model: foreign Key

primary key

Öğrenci Tablosu

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

foreign key

Dersler Tablosu

öğrencino	derskodu	kredi	notu
17532	BL44	4	85
17327	MK25	3	73
17532	BL13	3	94
17236	T121	2	45
16458	KM32	1	52

Kırmızı okla belirtilen 17532 nolu öğrencinin farklı derslere ait iki adet kaydı bulunmaktadır

Relational Data Model: foreign Key

primary key

Öğrenci Tablosu

öğrencino	isim	kullanıcıadı	yaş	ortalama
17532	Eymen	Eymen234	19	3.2
17327	Mustafa	m.4417	18	4
17347	Kemal	Kml45	17	3.7
17236	Cemil	Cm3418	18	2.9
16458	Hayri	Hayri95	19	3.8

foreign key

Dersler Tablosu

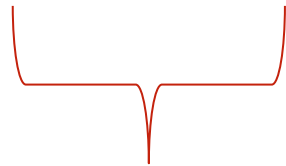
öğrencino	derskodu	kredi	notu
17532	BL44	4	85
17327	MK25	3	73
17532	BL13	3	94
17236	T121	2	45
16458	KM32	1	52

- Öğrenci tablosundan bulunmayan bir kaydı ekleyemezsiniz
- Dersler tablosunda kaydı bulunan bir öğrenciyi Öğrenci tablosundan silemezsiniz
- Bu kısıtlamalara ek olarak SQL dilini kullanarak tabloları oluştururken ek kısıtlamalar eklenebilir.
 - yaş alanı 17 den büyük olarak tanımlanabilir

Relational Data Model: Composit Key

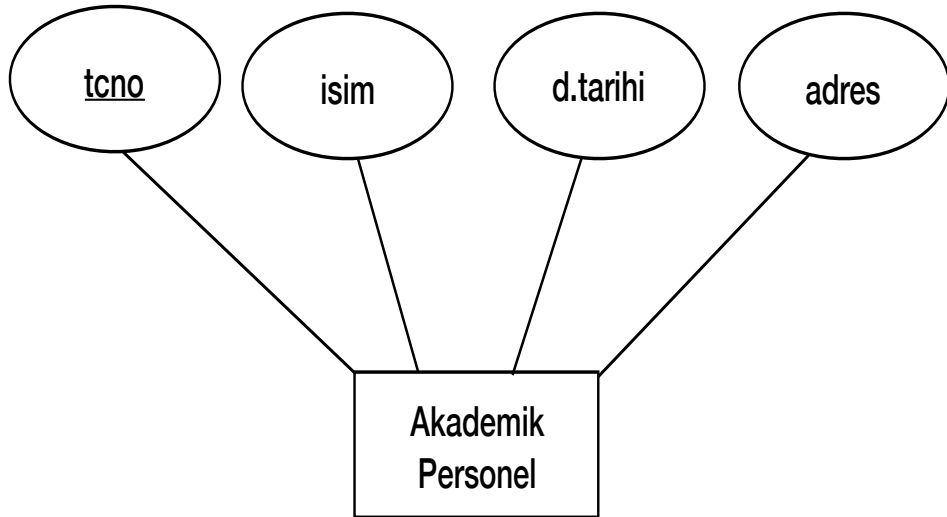
Birden fazla alanın bir araya getirilerek oluşturduğu anahtara primary key denir.

müşterino	siparişno	ürünkodu	ürünsayısı
C01	O001	P007	23
C02	O123	P007	19
C02	O123	P230	82
C01	O001	P890	42



{müşterino, siparişno}

ER Model'den ilişkisel Modele

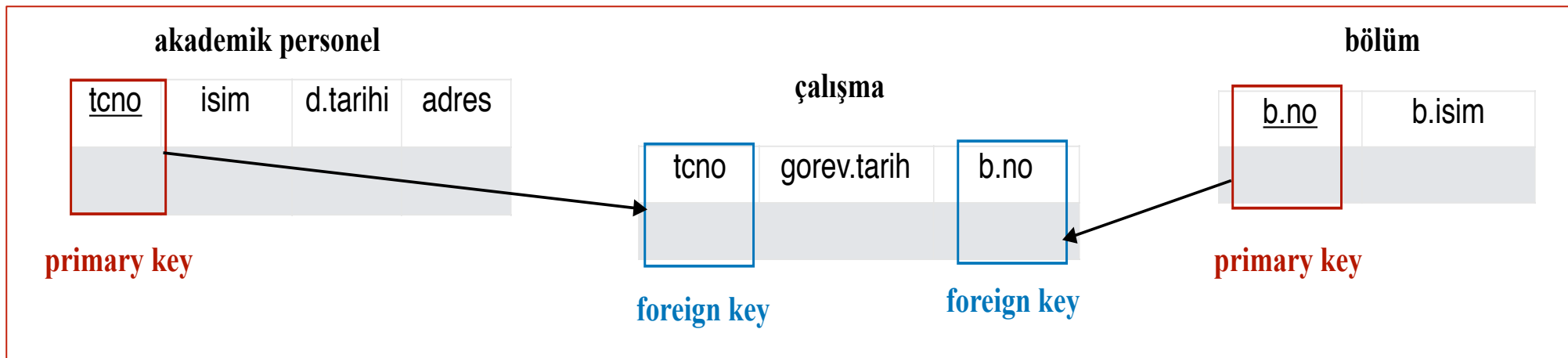
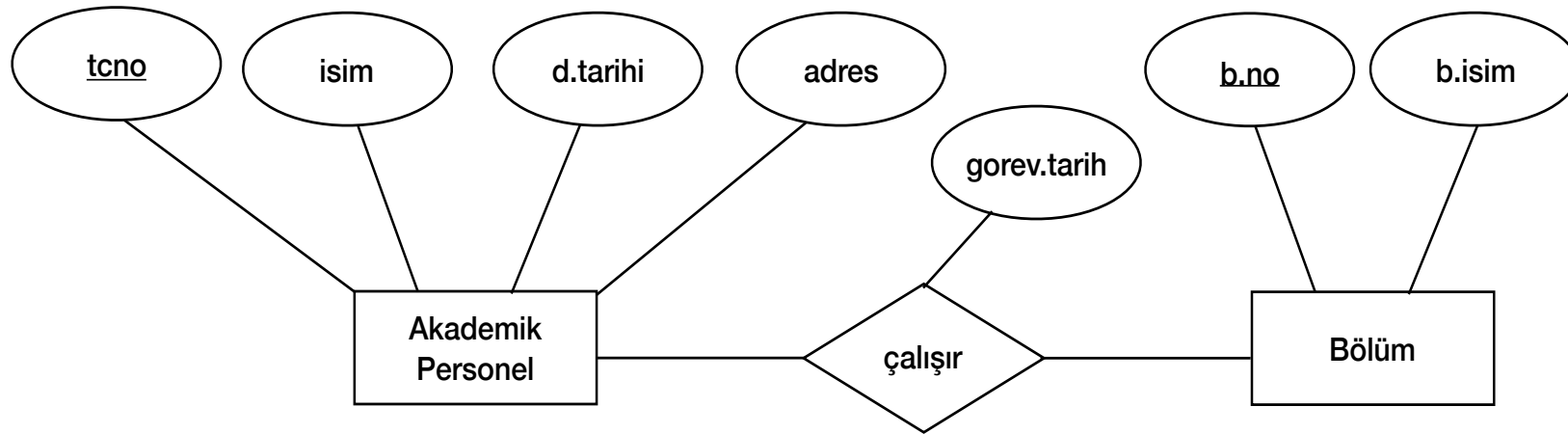


Akademik Personel
ER diagramı

Akademik Personel
tablosunu oluşturalım

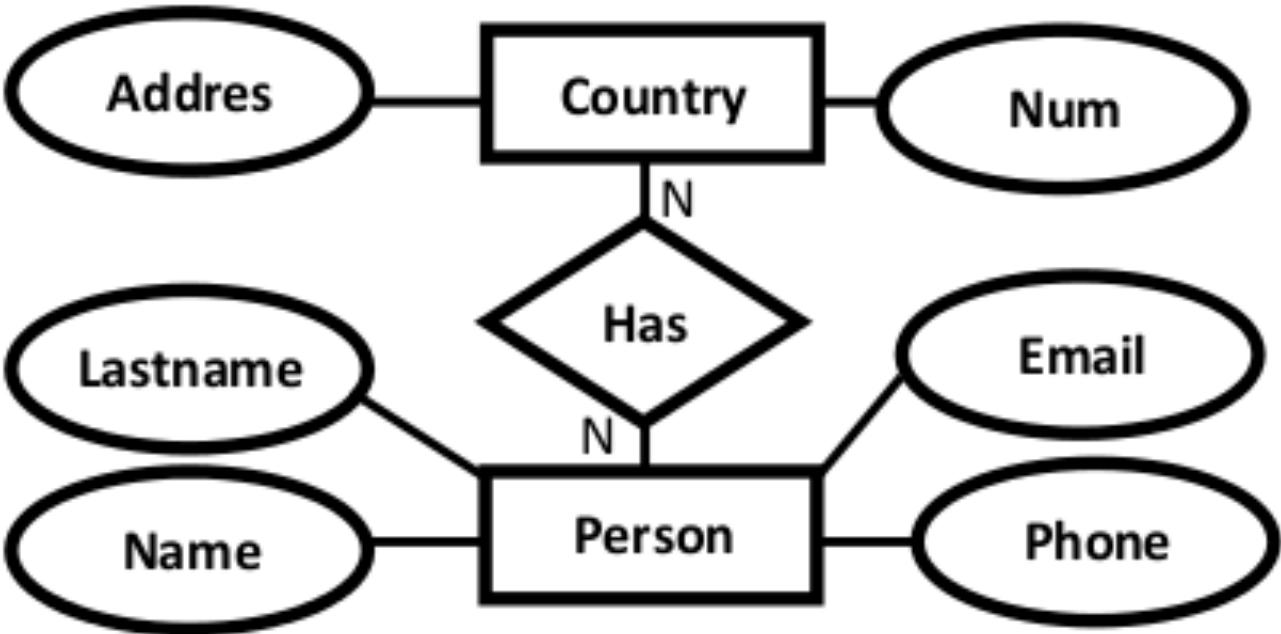
<u>tcno</u>	isim	d.tarihi	adres
x	y	z	†
.	.	.	.

ER Model'den ilişkisel Modele





Örnek



İlişkisel Cebir (Relational Algebra)

- İlişkisel cebir, ilişkisel veri modelinde bulunan bir **biçimsel sorgulama dilidir**.
- Tablolara kaydedilecek olan verilerin sorgularken oluşturulan matematiksel tanımını içerir.
- İlişkisel cebirde bulunan bir sorgu işlemsel operatörlerden oluşmaktadır.

İlişkisel Cebir: Operatörler

- Bu operatörler aynı anda bir ya da iki ilişkiyi hesaplamada kullanılır.
- İlişkisel cebir ve operatörleri yardımıyla veritabanına yönlendirilecek sorgu sırasıyla ve aşamalar halinde tamamlanır.

İlişkisel Cebir: Operatörler

- Seçim (Selection- σ)
- İzdüşüm-Yansıtma (Projection- π)
- Kartezyen Çarpımı (Cross-product- \times)
- Fark (Difference - $-$)
- Birleşim (*union*- \cup)
- Şartlı Bitiştirme (Conditional Join - $\bowtie_{\text{şart}}$)
- Eşit Bitiştirme (Equijoin - \bowtie_e)
- Bölme (*division* - $/$)
- Kesişim (intersection - \cap)

Tablolar

Kitap

kitap.no	yayinevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Yayinevi

yayinevi.no	yayinevi.adı	lokasyon	tel
235	MF	İstanbul	4244
121	YF	Malatya	5454
222	UF	Ankara	7243
567	ZF	Elazığ	4423

Kitap2

kitap.no	yayinevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

İlişkisel Cebir: Seçim

Seçim (Selection- σ) Verilen tabloyu kullanarak seçim şartını sağlayan satırları seçer.

$$\sigma_{\text{reyting} > 8} (\text{Kitap})$$

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

kitap.no	yayınevi.no	kitap.adı	reyting
4	222	U	10

İlişkisel Cebir: Seçim

Seçim (Selection- σ)

$\sigma_{\text{reyting} < 6} (\text{Kitap})$

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
3	121	T	4

İlişkisel Cebir: İzdüşüm

İzdüşüm-Yansıtma (Projection- π)

İlişkinin istenmeyen sütunlarını siler.

Yayınevi

$\pi_{yayınevi.no, lokasyon} (Yayınevi)$

yayınevi.no	lokasyon
235	İstanbul
121	Malatya
222	Ankara
567	Elazığ

yayınevi.no	yayınevi.adı	lokasyon	tel
235	MF	İstanbul	4244
121	YF	Malatya	5454
222	UF	Ankara	7243
567	ZF	Elazığ	4423

İlişkisel Cebir: İzdüşüm

İzdüşüm-Yansıtma (Projection- π)

ilişkinin istenmeyen sütunlarını siler.

Kitap

$\pi_{\text{kitap.adı, reyting}} (\sigma_{\text{reyting} > 7} (\text{Kitap}))$

kitap.adı	reyting
Y	8
U	10

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

İlişkisel Cebir: Kartezyen Çarpımı

Kartezyen Çarpımı (Cross-product- X) Her bir tablonun satırı diğer tablonun satırları ile çarpılacaktır

ρ (C($1 \rightarrow \text{kitap.no1}$,
 $2 \rightarrow \text{yayınevi.no1}$,
 $3 \rightarrow \text{kitap.adı1}$,
 $4 \rightarrow \text{reyting1}$), Kitap \times Kitap2)

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

İlişkisel Cebir: Kartezyen Çarpımı

Kartezyen Çarpımı (Cross-product- X) Her bir tablonun satırı diğer tablonun satırları ile çarpılacaktır

ρ (C($1 \rightarrow \text{kitap.no1}$, $2 \rightarrow \text{yayınevi.no1}$, $3 \rightarrow \text{kitap.adı1}$, $4 \rightarrow \text{reyting1}$), $\text{Kitap} \times \text{Kitap2}$)

kitap.no1	yayınevi.no1	kitap.adı1	reyting1	kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5	1	235	X	5
1	235	X	5	8	121	M	10
1	235	X	5	11	567	S	8
2	235	Y	8	1	235	X	5
2	235	Y	8	8	121	M	10
2	235	Y	8	11	567	S	8
..		.					
4	222	U	10	11	567	S	8

İlişkisel Cebir: Fark

Fark (Difference - -) Birinci ilişkide bulunup ikinci ilişkide bulunmayan değerleri verir.

Kitap – *Kitap2*

kitap.no	yayınevi.no	kitap.adı	reyting
2	235	Y	8
3	121	T	4
4	222	U	10

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

İlişkisel Cebir: Yeniden Adlandırma

(renaming- ρ): Alanların adlarını yeniden belirlemeyi sağlar.

$\rho (C(1 \rightarrow \text{kitap.no1}, 2 \rightarrow \text{yayınevi.no1}, 3 \rightarrow \text{kitap.adı1}, 4 \rightarrow \text{reyting1}), \text{Kitap} \chi \text{Kitap2})$

İlişkisel Cebir: Birleşim

Birleşim (*union*- \cup)

iki ilişkinin birleşimini verir.

Kitap \cup Kitap2

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10
8	121	M	10
11	567	S	8

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

İlişkisel Cebir: Şartlı Bitiştirme

Şartlı Bitiştirme (Conditional Join - ⋈_{şart})

Kitap ⋈_{Kitap.reyting > Kitap2.reyting} Kitap2

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

kitap.no1	yayınevi.no1	kitap.adı1	reyting1	kitap.no	yayınevi.no	kitap.adı	reyting
2	235	Y	8	1	235	X	5
4	222	U	10	1	235	X	5
4	222	U	10	11	567	S	8

İlişkisel Cebir: Eşit Bitiştirme

Eşit Bitiştirme (Equijoin - ⋈_e)

Tablolarda bulunan şartı sağlayan alanların kartezyen çarpımını verir

Kitap ⋈_{Kitap.kitap.no= Kitap2.kitap.no} Kitap2

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

kitap.no	yayınevi.no1	kitap.adı1	reyting1	yayınevi.no	kitap.adı	reyting
1	235	X	5	235	X	5

İlişkisel Cebir: Doğal Bitiştirme

Doğal Bitiştirme (Naturaljoin - ⋈) iki tabloda bulunan alanlardan birebir örtüşenleri listeler

Kitap ⋈ Kitap2

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

İlişkisel Cebir: Bölme

Bölme (*division* - /)

İki tablo arasındaki bir tabloda bulunan değer 2.Tablonun bütün verilerini içeriyorsa kullanılır.

müşteriadi	ürünadı
Asım	Tablet
Ali	Telefon
İsmet	Bilgisayar
Kamil	Tablet
Asım	Telefon
Ali	Bilgisayar
Tuna	Telefon
Asım	Bilgisayar

ürünno	ürünadı
1	Tablet
2	Telefon
3	Bilgisayar

müşteri / ürün

bir firmanın ürün tablosundaki ürünlerin hepsini alan müşterileri bulmak için bölme kullanılır.

müşteriadi
Asım

İlişkisel Cebir: Kesişim

Kesişim (intersection \cap)

iki ilişkide bulunan ortak satırları verir

$Kitap \cap Kitap2$

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
2	235	Y	8
3	121	T	4
4	222	U	10

Kitap

kitap.no	yayınevi.no	kitap.adı	reyting
1	235	X	5
8	121	M	10
11	567	S	8

Kitap2

İlişkisel Cebir

- Birleşim, kesişim ve fark işlemlerini gerçekleştirmek için aynı alan sahip olan tablolar kullanılmalıdır.
- İki den fazla tablonun birleştirilmesi işlemleri için join operatörleri kullanılmaktadır.
- Birleştirme işlemi iki tablonun belirlenen şartlar çerçevesindeki kartezyen çarpımının sonucudur.
- Bu işlem yapılırken gereken seçim veya izdüşüm gibi işlemler gerçekleştirilebilir.