

Veritabanı Yönetim Sistemleri

(Veritabanı Tasarımı)

İlişkisel Cebir

Celal ÇEKEN ve Veysel Harun ŞAHİN



Konular

- ✓ Biçimsel Sorgulama Dilleri
- ✓ İlişkisel Cebir
- ✓ İlişkisel Cebir İşlemleri
- ✓ Seçme (select) işlemi
- ✓ Projeksiyon (projection) işlemi
- ✓ Birleşim (union) işlemi
- ✓ Kesişim (intersect) işlemi
- ✓ Fark (difference) işlemi
- ✓ Kartezyen çarpım (cartesian product) işlemi
- ✓ Birleştirme (join) işlemi
- ✓ Bölme (division) işlemi
- ✓ Uygulamalar

Biçimsel Sorgulama Dilleri

- ✓ İlişkisel modelde sıklıkla kullanılan iki biçimsel sorgulama dili vardır: İlişkisel Cebir (Relational Algebra) ve İlişkisel Hesap (Relational Calculus).
 - Bu diller biçimseldir. Herhangi bir yorumlayıcı veya derleyici yoktur.
 - Bu diller, SQL (Structured Query Language) ve QBE (Query By Example) dillerinin temelini oluştururlar.
- ✓ **İlişkisel Cebir (Relational Algebra):** Prosedürelidir (procedural) ve sorguların hangi adımlardan oluştuğu açıkça ifade edilir.
 - Öğrenciler ve İller tablosunu birleştir.
 - İli Sakarya olan öğrencileri seç.
 - Bu öğrencilerin numara, ad ve soyad bilgilerini göster.

Biçimsel Sorgulama Dilleri

- ✓ **İlişkisel Hesap (Relational Calculus):** Prosedürel değildir. Bildirimseldir (declarative). Kullanıcıların ne istediklerini bildirmelerine olanak tanır. Nasıl hesaplanması gerektiği bildirilmez.
 - Sakarya’da kayıtlı öğrencilerin, numara, ad ve soyadı bilgilerini göster.

İlişkisel Cebir

- ✓ İlişkisel Veritabanı Yönetim Sistemlerinde temel yapı ilişkidir (ilişki yerine tablo ifadesi daha çok kullanılır). İlişkisel modelde kullanıcılar tablolar cinsinden düşünür, tablolar üzerinde işlem yaparlar ve sonuçlar da tablo yapısında elde edilir.
- ✓ İlişkisel cebir, biçimsel sorgulama dilidir.
- ✓ İlişkisel cebir yardımıyla, veritabanının nasıl sorgulanabileceği yorumlanır.
- ✓ Sorgular SQL'den farklı olarak biçimseldir. Herhangi bir yorumlayıcı veya derleyici yoktur.

İlişkisel Cebir

- ✓ Seçme (select) işlemi
- ✓ Projeksiyon (projection) işlemi
- ✓ Birleşim (union) işlemi
- ✓ Kesişim (intersect) işlemi
- ✓ Fark (difference) işlemi
- ✓ Kartezyen çarpım (cartesian product) işlemi
- ✓ Birleştirme (join) işlemi
- ✓ Bölme (division) işlemi

Seçme (Select) İşlemi

- ✓ Bir tablodan bir veya daha fazla kritere göre kayıtları seçme işlemidir.
- ✓ Gösterim: $\sigma_{\text{seçim kriteri}}(\text{TabloAdı})$
- ✓ Seçim kriterlerinde kullanılan semboller: \wedge (ve), \vee (veya), \neg (değil)
- ✓ $\sigma_{\text{ogrenciNo} = '1512B10010'}(\text{Ogrenciler})$: Ogrenciler tablosundan ogrenciNo bilgisi 1512B100010 olan öğrenciyi seç.
- ✓ $\sigma_{\text{cinsiyet} = 'E' \wedge \text{yas} > 18}(\text{Kisiler})$: Kisiler tablosundan cinsiyeti Erkek olan ve yaşı 18'den büyük olan kayıtları seç.

Projeksiyon (Projection) İşlemi

- ✓ Bir tablodan yalnızca belirli niteliklere ait bilgiler seçilerek gösterilir.
- ✓ Tekrar eden satırlar bir kere gösterilir.
- ✓ Gösterim: $\Pi_{\text{nitelik listesi}}(\text{TabloAdı})$
- ✓ Birden fazla nitelik varsa nitelikler birbirlerinden virgül ile ayrılır.
- ✓ $\Pi_{\text{ogrenciNo}, \text{ad}, \text{soyad}}(\text{Ogrenciler})$: Ogrenciler tablosundaki tüm kayıtların ogrenciNo, ad ve soyad bilgilerini göster.
- ✓ $\Pi_{\text{ad}, \text{soyad}}(\sigma_{\text{fakulteNo} = 12}(\text{Ogrenciler}))$: Ogrenciler tablosundan fakülte numarası 12 olan tüm kayıtların ad ve soyad bilgilerini göster.

Birleşim (Union) İşlemi

- ✓ İki tablonun küme birleşimi alınarak gösterilir.
- ✓ Tabloların nitelik sayısı aynı olmalı.
- ✓ Tabloların aynı sıradaki niteliklerinin değer alanları aynı olmalı.
- ✓ Birleşim işlemi sonrasında birbirinin aynısı olan satırlar bir kez getirilir.
- ✓ Gösterim: $TabloAdı1 \cup TabloAdı2$
- ✓ $\Pi_{ogrenciNo, ad, soyad}(BilgisayarToplulugu) \cup \Pi_{ogrenciNo, ad, soyad}(SiberToplulugu)$
 - BilgisayarToplulugu ve SiberToplulugu tablolarındaki kayıtların ogrenciNo, ad ve soyad alanlarındaki bilgileri birleştir ve göster.

Kesişim (Intersect) İşlemi

- ✓ İki tablonun küme kesişimi alınarak gösterilir.
- ✓ Tabloların nitelik sayısı aynı olmalı.
- ✓ Tabloların aynı sıradaki niteliklerinin değer alanları aynı olmalı.
- ✓ Gösterim: $TabloAdı1 \cap TabloAdı2$
- ✓ $\Pi_{ogrenciNo, ad, soyad}(BilgisayarToplulugu) \cap \Pi_{ogrenciNo, ad, soyad}(SiberToplulugu)$
 - BilgisayarToplulugu ve SiberToplulugu tablolarındaki kayıtların ogrenciNo, ad ve soyad alanlarındaki bilgileri aynı olan kayıtların bu alanlardaki bilgilerini seç ve göster.

Fark (Difference) İşlemi

- ✓ İki tablonun küme farkı alınarak gösterilir.
- ✓ Tabloların nitelik sayısı aynı olmalı.
- ✓ Tabloların aynı sıradaki niteliklerinin değer alanları aynı olmalı.
- ✓ Gösterim: $TabloAdı1 - TabloAdı2$
- ✓ $\Pi_{ogrenciNo, ad, soyad}(BilgisayarToplulugu) - \Pi_{ogrenciNo, ad, soyad}(SiberToplulugu)$
 - BilgisayarToplulugu ve SiberToplulugu tablolarındaki kayıtların ogrenciNo, ad ve soyad alanlarındaki bilgileri seç ve bu bilgilere göre karşılaştırma yap.
 - BilgisayarTopluluğu tablosunda olup da SiberToplulugu tablosunda olmayan öğrencilerin bu bilgilerini göster.

Kartezyen Çarpım (Cartesian Product) İşlemi

- ✓ İki tablonun kartezyen çarpımı gerçekleştirilir.
- ✓ Farklı iki tabloya uygulanır.
- ✓ Gösterim: $TabloAdı1 \times TabloAdı2$

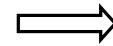
Muzisyenler

<u>muzisyenNo</u>	adi	soyadi	ilKodu
9	Ayşe	Yılmaz	00
12	Mehmet	Yorulmaz	06
15	Merve	Sakar	00
20	Kağan	Yalın	06

X

İller

<u>ilKodu</u>	ilAdı
00	Bilinmiyor
01	Adana
06	Ankara



muzisyenNo	adi	soyadi	ilKodu	ilKodu	ilAdı
9	Ayşe	Yılmaz	00	00	Bilinmiyor
9	Ayşe	Yılmaz	00	01	Adana
9	Ayşe	Yılmaz	00	06	Ankara
12	Mehmet	Yorulmaz	06	00	Bilinmiyor
12	Mehmet	Yorulmaz	06	01	Adana
12	Mehmet	Yorulmaz	06	06	Ankara
15	Merve	Sakar	00	00	Bilinmiyor
15	Merve	Sakar	00	01	Adana
15	Merve	Sakar	00	06	Ankara
20	Kağan	Yalın	06	00	Bilinmiyor
20	Kağan	Yalın	06	01	Adana
20	Kağan	Yalın	06	06	Ankara

Doğal Birleştirme/İç Birleştirme (Natural Join/Inner Join) İşlemi

- ✓ İki tablonun kartezyen çarpımının alt kümesi elde edilir.
- ✓ Kartezyen çarpım soncuna seçme işlemi uygulanır.
- ✓ Aynı niteliğe ait bilgileri aynı olan kayıtlar seçilerek gösterilir.
- ✓ Gösterim: $TabloAdı1 \bowtie TabloAdı2$

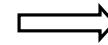
Muzisyenler

<u>muzisyenNo</u>	adi	soyadi	ilKodu
9	Ayşe	Yılmaz	00
12	Mehmet	Yorulmaz	06
15	Merve	Sakar	00
18	Hale	Çınar	54
20	Kağan	Yalın	06



İller

<u>ilKodu</u>	ilAdı
00	Bilinmiyor
01	Adana
06	Ankara
34	İstanbul



muzisyenNo	adi	soyadi	ilKodu	ilKodu	ilAdı
9	Ayşe	Yılmaz	00	00	Bilinmiyor
12	Mehmet	Yorulmaz	06	06	Ankara
15	Merve	Sakar	00	00	Bilinmiyor
20	Kağan	Yalın	06	06	Ankara

Sol Dış Birleştirme (Left Outer Join) İşlemi

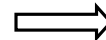
- ✓ İki tabloya uygulanır.
- ✓ Sol taraftaki tablodaki tüm kayıtlar alınır.
- ✓ Sağ taraftaki tablodan aynı niteliğe ait bilgileri aynı olan kayıtlar seçilir, bilgileri sol taraftaki tablodan alınan bilgilere eklenerek gösterilir.
- ✓ Sağ taraftaki tabloda eşleşen kayıt yoksa bunların yeri boş bırakılır (NULL).
- ✓ Gösterim: $TabloAdı1 \bowtie TabloAdı2$

Muzisyenler

<u>muzisyenNo</u>	adi	soyadi	ilKodu
9	Ayşe	Yılmaz	33
12	Mehmet	Yorulmaz	06
15	Merve	Sakar	00
20	Kağan	Yalın	06
22	Cenk	Dur	07

İller

<u>ilKodu</u>	ilAdı
00	Bilinmiyor
01	Adana
06	Ankara



muzisyenNo	adi	soyadi	ilKodu	ilKodu	ilAdı
9	Ayşe	Yılmaz	33	NULL	NULL
12	Mehmet	Yorulmaz	06	06	Ankara
15	Merve	Sakar	00	00	Bilinmiyor
20	Kaan	Yalın	06	06	Ankara
22	Cenk	Dur	07	NULL	NULL

Sağ Dış Birleştirme (Right Outer Join) İşlemi

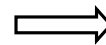
- ✓ İki tabloya uygulanır.
- ✓ Sağ taraftaki tablodaki tüm kayıtlar alınır.
- ✓ Sol taraftaki tablodan aynı niteliğe ait bilgileri aynı olan kayıtlar seçilir, bilgileri sağ taraftaki tablodan alınan bilgilere eklenerek gösterilir.
- ✓ Sol taraftaki tabloda eşleşen kayıt yok ise bunların yeri boş bırakılır (NULL).
- ✓ Gösterim: $TabloAdı1 \bowtie TabloAdı2$

Muzisyenler

<u>muzisyenNo</u>	adi	soyadi	ilKodu
9	Ayşe	Yılmaz	00
12	Mehmet	Yorulmaz	06
15	Merve	Sakar	00
20	Kağan	Yalın	06

İller

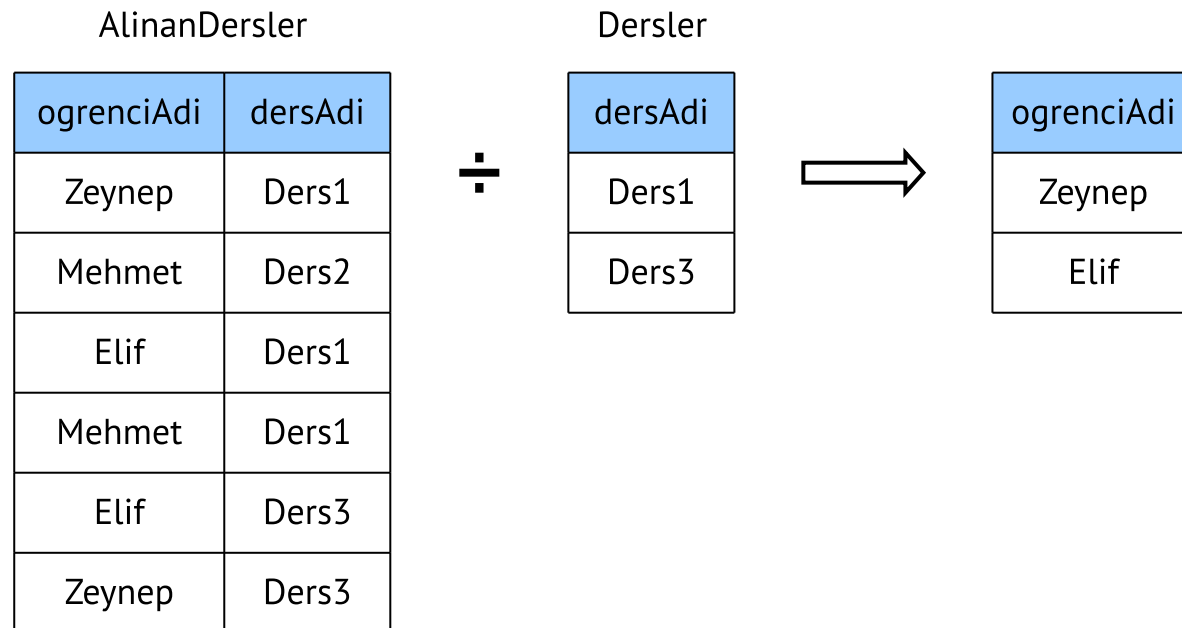
<u>ilKodu</u>	ilAdı
00	Bilinmiyor
01	Adana
06	Ankara



muzisyenNo	adi	soyadi	ilKodu	ilKodu	ilAdı
9	Ayşe	Yılmaz	00	00	Bilinmiyor
15	Merve	Sakar	00	00	Bilinmiyor
NULL	NULL	NULL	NULL	01	Adana
12	Mehmet	Yorulmaz	06	06	Ankara
20	Kaan	Yalın	06	06	Ankara

Bölme (Divison) İşlemi

- ✓ İki tabloya uygulanır.
- ✓ İki tablonun bir ortak niteliği olması gerekir.
- ✓ Gösterim: $TabloAdı1 \div TabloAdı2$



Uygulama

- ✓ 3. Hafta ders notlarında geliştirilen Üniversite Bilgi Sistemine ait veritabanının ilişkisel şeması (alt şema) aşağıdaki gibidir. İstenen sorgulara ait biçimsel ifadeleri ilişkisel cebir ile yapınız.
- ✓ Bolum (bolumNo: int, ad: varchar(20))
- ✓ Ogrenci (ogrenciNo: char(10), ad: varchar(20), soyad: varchar(20), dogumTarihi: date, dogumYeri: varchar(16), cinsiyet: char(1), bolumNo: int)
- ✓ AcilanDers (dersNo: char(6), dersKodu: char(5), donem: varchar(10), sicilNo: char(7), bolumNo: int)
- ✓ Kayit (kayitNo: int, dersNo: char(6), ogrenciNo: char(10), vize: numeric, final: numeric, ortalama: numeric)
- ✓ OgretimUyesi (sicilNo: char(7), ad: varchar(20), soyad: varchar(20), dogumTarihi: date, dogumYeri: char(16), unvan: varchar(20))

Uygulama

- ✓ Soru 1: Bütün öğrencilerin öğrenci numarası, ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.
- ✓ Soru 2: Sakarya ilinde doğan öğrencilerin adını ve soyadını listeleyen ilişkisel cebir ifadesini yazınız.
- ✓ Soru 3: Bilgisayar Mühendisliği bölümünde ders veren öğretim üyelerinin ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.
- ✓ Soru 4: 2016-2017 eğitim öğretim döneminde hiçbir dersi olmayan öğretim üyelerinin sicil numarası, ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.
- ✓ Soru 5: Makine Mühendisliği bölümünde okuyan veya cinsiyeti kız olan öğrencilerin ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.
- ✓ Soru 6: Çevre Mühendisliği bölümünde okuyan ve not ortalaması 3'ün üzerinde olan öğrencilerin öğrenci numarası bilgilerini listeleyen ilişkisel cebir ifadesini yazınız.

Uygulama

- ✓ Soru1: Bütün öğrencilerin öğrenci numarası, ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.

$\Pi_{ogrenciNo, ad, soyad}(Ogrenci)$

Uygulama

- ✓ Soru 2: Sakarya ilinde doğan öğrencilerin adını ve soyadını listeleyen ilişkisel cebir ifadesini yazınız.

$$\Pi_{ad, soyad}(\sigma_{dogumYeri = 'Sakarya'}(Ogrenci))$$

Uygulama

- ✓ Soru 3: Bilgisayar Mühendisliği bölümünde ders veren öğretim üyelerinin ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.

$$\Pi_{ad, soyad}(OgretimUyesi \bowtie AcilanDers \bowtie (\sigma_{ad = 'Bilgisayar Mühendisliği'}(Bolum)))$$

Uygulama

- ✓ Soru 4: 2016-2017 eğitim öğretim döneminde hiçbir dersi olmayan öğretim üyelerinin sicil numarası, ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.

$\Pi_{\text{sicilNo, ad, soyad}}(\text{OgretimUyesi})$

– $\Pi_{\text{sicilNo, ad, soyad}}(\text{OgretimUyesi} \bowtie (\sigma_{\text{donem} = '2016-2017'}(\text{AcilanDers})))$

Uygulama

- ✓ Soru 5: Makine Mühendisliği bölümünde okuyan veya cinsiyeti kız olan öğrencilerin ad ve soyadlarını listeleyen ilişkisel cebir ifadesini yazınız.

$$\Pi_{ad, soyad}(Ogrenci \bowtie \Pi_{bolumNo}(\sigma_{ad = 'Makine Mühendisliği'}(Bolum))) \\ \cup \Pi_{ad, soyad}(\sigma_{cinsiyet = 'Kız'}(Ogrenci)))$$

Uygulama

- ✓ Soru 6: Çevre Mühendisliği bölümünde okuyan ve not ortalaması 3'ün üzerinde olan öğrencilerin öğrenci numarası bilgilerini listeleyen ilişkisel cebir ifadesini yazınız.

$$\Pi_{ogrenciNo}(Ogrenci \bowtie \Pi_{bolumNo}(\sigma_{ad = 'Çevre Mühendisliği'}(Bolum))) \\ \cap \Pi_{ogrenciNo}(Ogrenci \bowtie \sigma_{ortalama > 3}(Kayit))$$