

Veritabanı Yönetim Sistemleri (335)

Yrd. Doç. Dr. Ahmet Arif AYDIN

December 17

Bu haftanın Konusu

| Konular |
|--------------------------------|
| Veritabanı Tasarımı |
| ER-Model |
| UML |
| Relational Model |
| İlişkisel Cebir |
| İlişkisel Hesap |
| SQL |
| Dosya Yapıları ve İndexleme |
| Sorgu Optimizasyonu |
| Hareket Yönetimi |
| Kilitlenmeler |
| Concurrency |
| Veritabanı Kurtarma Teknikleri |

- Relational Calculus
 - Tuple Relational Calculus
 - Domain Relational Calculus
- SQL

Logic: Mantık

- VTYS'lerin sorgulama işleminin temelinde Mantık bulunmaktadır.
- Sorguları ve kısıtlamaları oluşturmak için Mantıksal ifadeler kullanılır
 - Logical Expressions - Relational Calculus Expressions
- İlişkisel veri tabanları küme tabanlıdır
 - Entity set
 - Relationship set

Transformation Rules

Dönüşüm kuralları

- Eşitlik
 - ifade1 eşittir ifade2
 - ifade1 \equiv ifade2
- Aynı
 - ifade1 ve ifade2 birebir aynı ise (identically equal) ikisinin manaları da aynıdır
- Yeniden yazma kuralları
 - Bir ifadede ifade1 kullanılmışsa aynı ifade ifade2 ile de tekrar yazılabilir.

Implication Law

Implication : gerektirmek, ima etmek

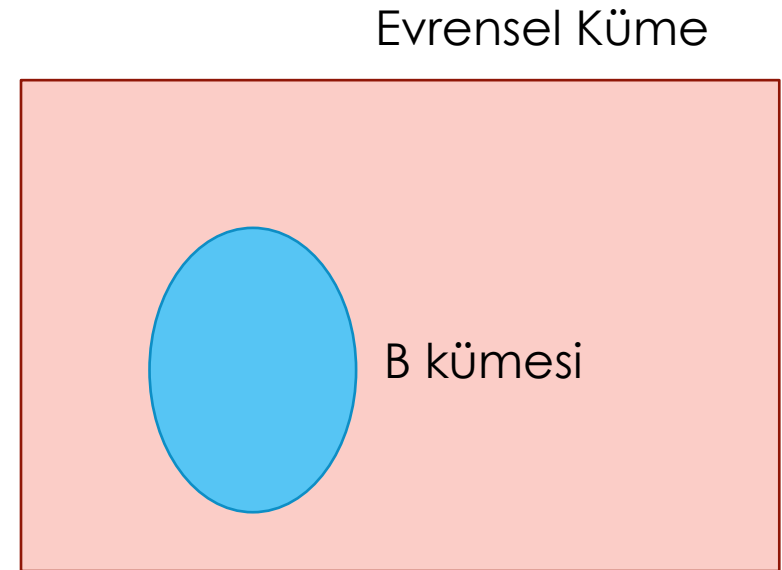
- $P \rightarrow Q$
 - If p then q \equiv (not p) or q
 - Eğer p doğru ise q da doğrudur (tersi doğru değildir)
- P = Hava kapalıdır
- Q = Güneş görünmemektedir

$P \rightarrow Q$ eşit değildir $Q \rightarrow P$

Double Negation Law

Bir ifadenin **değilinin değili** (olumsuzun olumsuzu) kendisidir.

- $\text{not}(\text{not } p) \equiv p$



De Morgan Kuralı

DeMorgan's Law

- $\text{not } (p \text{ and } q) \equiv (\text{not } p) \text{ or } (\text{not } q)$
- $\text{not } (p \text{ or } q) \equiv (\text{not } p) \text{ and } (\text{not } q)$

Distributive Law

Dağıtım Kuralı

- $p \text{ and } (q \text{ or } r) \equiv (p \text{ and } q) \text{ or } (p \text{ and } r)$
- $p \text{ or } (q \text{ and } r) \equiv (p \text{ or } q) \text{ and } (p \text{ or } r)$

İlişkisel Hesap (Relational Calculus)

- İlişkisel cebirin (relational algebra) bir alternatifidir.
- Declerativedir.
- Cevapları nasıl hesaplanacağını detayları verilmez (nonprocedural)
- SQL dilinin ortaya çıkmasında büyük etkisi olmuştur.

İlişkisel Hesap (Relational Calculus)

- Demet İlişkisel Hesap (Tuple Relational Calculus (TRC))
- Alan ilişkisel Hesap (Domain Relational Calculus (DRC)

Tuple Relational Calculus (TRC)

- $\{ t \mid P(t) \}$ veya $\{ t \mid \text{şart}(t) \}$ $t = \underline{\text{şartı sağlayan tuple'lar}}$ (tabloda bulunan satırlar)
- $\{ t \mid \text{işçi}(t) \text{ and } t.\text{yaş} > 40 \}$
 - Yaşı 40 dan büyük olan işçilerin hepsinin bütün bilgilerini (tuple) getir
- $\{ t \mid \text{personel}(t) \text{ and } t.\text{bölümno} = 10 \}$
 - 10 nolu bölümde çalışan personelin hepsinin bütün bilgilerini (tuple) getir

Tuple Relational Calculus (TRC)

- $\{ t \mid \text{şart}(t) \}$
- Şart:
 - Atomik formül
 - $\neg P = \text{not } P$ (p nin değili)
 - $P \wedge Q = P \text{ ve } Q$
 - $(P \vee Q) = P \text{ veya } Q$
 - $P \rightarrow Q = P \text{ doğru ise } Q \text{ da doğrudur}$
 - $\exists R (\text{şart} (R)) : (\exists : \text{en az bir tane var(**there exists**)})$
 - $\forall K (\text{şart} (K)) \quad (\forall : \text{hepsi için (for all)})$

Domain Relational Calculus

Domain Relational Calculus (DRC)

- $\{ \langle \text{sütun1}, \text{sütun2}, \text{sütun3}, \dots \text{sütunn} \rangle \mid P(\text{sütun1}, \text{sütun2}, \text{sütun3}, \dots \text{sütunn}) \}$
- Hesaplanan sonuc: tuple(satır) yerine seçilen sütunlardır.
- $\{ \langle \text{işçi.no}, \text{işçi.adı} \rangle \mid \langle \text{işçi.no}, \text{işçi.adı} \rangle ? \text{İşçi} \wedge \text{bölüm.no} = 10 \}$
 - 10 nolu bölümde çalışan işçilerin adını ve işçi nosunu getir.

Örnek

KullanıcıReyting Tablosu

| kullanıcıno | isbn | reyting |
|-------------|------------|---------|
| 276746 | 0553561618 | 0 |
| 276746 | 055356451X | 2 |
| 276746 | 0451166892 | 3 |
| 276746 | 0786014512 | 0 |
| 276747 | 0060517794 | 9 |
| 276747 | 0451192001 | 0 |
| 276751 | 3596218098 | 8 |
| 276754 | 0684867621 | 8 |
| 276755 | 0451166892 | 5 |

University of California Irvine

<http://archive.ics.uci.edu/ml/index.php>

278,858 kullanıcı
1,149,780 reyting
271,379 kitaplar

Örnek

Reytingi 0 olan kitapları bul

$\{ t \mid \text{şart}(t) \}$

$\{ T \mid K \text{ E KullanıcıReyting} \wedge K. \text{reyting} = 0 \}$



| kullanıcıno | isbn | reyting |
|-------------|------------|---------|
| 276746 | 0553561618 | 0 |
| 276746 | 055356451X | 2 |
| 276746 | 0451166892 | 3 |
| 276746 | 0786014512 | 0 |
| 276747 | 0060517794 | 9 |
| 276747 | 0451192001 | 0 |
| 276751 | 3596218098 | 8 |
| 276754 | 0684867621 | 8 |
| 276755 | 0451166892 | 5 |

Örnek

Reytingi 5 den büyük olan kitapların isbn numaralarını ve reytingi veren kullanıcıno alanlarını getir.

{ t | şart(t) }

{ T | \exists K \in KullanıcıReyting

(K.reyting > 5 /\

Tkullanıcı = K.kullanıcıno /\

Tisbn = K.isbn)}

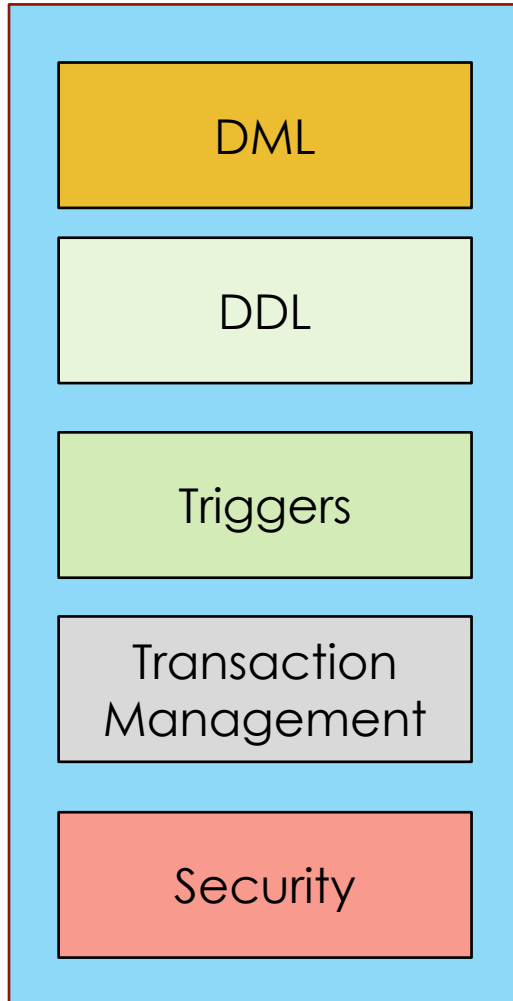
| Tkullanıcı | Tisbn |
|------------|------------|
| 276747 | 0060517794 |
| 276751 | 3596218098 |
| 276754 | 0684867621 |



| kullanıcıno | isbn | reyting |
|-------------|------------|---------|
| 276746 | 0553561618 | 0 |
| 276746 | 055356451X | 2 |
| 276746 | 0451166892 | 3 |
| 276746 | 0786014512 | 0 |
| 276747 | 0060517794 | 9 |
| 276747 | 0451192001 | 0 |
| 276751 | 3596218098 | 8 |
| 276754 | 0684867621 | 8 |
| 276755 | 0451166892 | 5 |

Structured Query Language

SQL



1970 tarihinde IBM tarafından geliştirilen ve RDBMS'lerde kullanılan bir standarttır

Data Manipulation Language: SQL'in satır ekle, sil , değiştir komutlarını içerir.

Data Definition Language: SQL tablolarının ve görünüşlerin oluşturulması, silinmesi, ve değiştirilmesi komutlarını içerir.

Triggers: Veritabanında yapılan bazı değişikliklerin yapılmasıyla tetiklenen işlemlerdir (action).

Transaction Management: Hareket yönetimi ile alakalı tanımlamaların yapıldığı kısımdır.

Security: SQL yardımıyla kullanıcıların veritabanı nesnelere erişim denetimi sağlanır.

SQL: Delete Table

öğrenci

| öğrencino | isim | kullanıcıadı | yaş | ortalama |
|-----------|---------|--------------|-----|----------|
| 17532 | Eymen | Eymen234 | 19 | 3.2 |
| 17327 | Mustafa | m.4417 | 18 | 4 |
| 17347 | Kemal | Km145 | 17 | 3.7 |
| 17236 | Cemil | Cm3418 | 18 | 2.9 |
| 16458 | Hayri | Hayri95 | 19 | 3.8 |

Bir tabloyu silmek için

```
DROP TABLE öğrenci;
```

```
CREATE TABLE öğrenci (  
    öğrencino int primary key,  
    isim varchar ,  
    kullanıcıadı varchar ,  
    yaş int,  
    ortalama real  
);
```

SQL: Kurallar

- SQL kodunda **spaces, tabs, and newlines** dikkate alınmaz
- '---' komut satırı ekler
- SQL case sensitive değildir.
 - `SELECT` , `Select` , `select`
- `varchar(80)`

SQL: Veri Tipleri

PosgreSQL standard SQL veri tiplerini destekler:

- smallint (2 bytes -32768 , + 32767)
- integer (4 bytes - 2147483648 to +2147483647)
- real
- double precision
- char(N)
- varchar(N)
- date
- time
- timestamp
- Interval
- PostgreSQL-specific data type

https://www.tutorialspoint.com/postgresql/postgresql_data_types.htm

SQL: Veri Tipleri

| Değer | Char (4) | Ayrılan Alan |
|------------|----------|--------------|
| '' | ' ' | 4 bytes |
| 'ab' | 'ab ' | 4 bytes |
| 'abcd' | 'abcd' | 4 bytes |
| 'abcdefgh' | 'abcd' | 4 bytes |

Char (n)
Ayrılan alan sabit

| Varchar(4) | Ayrılan Alan |
|------------|--------------|
| '' | 1 byte |
| 'ab' | 3 bytes |
| 'abcd' | 5 bytes |
| 'abcd' | 5 bytes |

Varchar (n),
Ayrılan alan değişken

<https://www.postgresql.org/docs/9.1/static/datatype-character.html>

SQL: insert into

| 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 |

Tabloya veri
ekleme komutu

1- **insert into** öğrenci **values** (17532, 'Eymen', 'Eymen234',19,3.2);

Tablonun bütün
kolonlarına
ekleme yapar

2- **insert into** öğrenci (öğrencino, isim, kullanıcıadı, yas, ortalama)
values (17532, 'Eymen', 'Eymen234',19,3.2);

Tablonun belirtilen kolonlarına ekleme yapar

SQL: Select

| 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 |

**Tabloda bulunan
veriyi listeler**

`Select * FROM öğrenci;`

SQL: Select

| 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 |

select öğrencino, isim **from** öğrenci;

| | ogrencino integer | isim character varying |
|---|----------------------|---------------------------|
| 1 | 17532 | Eymen |
| 2 | 17327 | Mustafa |
| 3 | 17347 | Kemal |
| 4 | 17236 | Cemil |
| 5 | 16458 | Hayri |

SQL: select

| 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 |

select * from öğrenci where yas>18;

| | ogrencino integer | isim character varying | kullanıcıadı character varying | yas integer | ortalama real |
|---|----------------------|---------------------------|-----------------------------------|----------------|------------------|
| 1 | 17532 | Eymen | Eymen234 | 19 | 3.2 |
| 2 | 16458 | Hayri | Hayri95 | 19 | 3.8 |

SQL: Where

| 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 |

select * from öğrenci where yas>17 and ortalama>3.0;

| | ogrencino integer | isim character varying | kullanıcıadı character varying | yas integer | ortalama real |
|---|----------------------|---------------------------|-----------------------------------|----------------|------------------|
| 1 | 17532 | Eymen | Eymen234 | 19 | 3.2 |
| 2 | 17327 | Mustafa | m.4417 | 18 | 4 |
| 3 | 16458 | Hayri | Hayri95 | 19 | 3.8 |

SQL: order

| 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 |

select * from öğrenci **order** by yas;

| | ogrencino integer | isim character varying | kullanıcıadı character varying | yas integer | ortalama real |
|---|----------------------|---------------------------|-----------------------------------|----------------|------------------|
| 1 | 17347 | Kemal | Kml45 | 17 | 3.7 |
| 2 | 17327 | Mustafa | m.4417 | 18 | 4 |
| 3 | 17236 | Cemil | Cm3418 | 18 | 2.9 |
| 4 | 17532 | Eymen | Eymen234 | 19 | 3.2 |
| 5 | 16458 | Hayri | Hayri95 | 19 | 3.8 |

SQL: distinct

| 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 |

select **distinct** yas from öğrenci order by yas;

| | yas integer |
|---|----------------|
| 1 | 17 |
| 2 | 18 |
| 3 | 19 |

SQL: Aggregate functions

Birden fazla satırdan sonuç olarak
sadece bir satır üreten fonksiyonlara aggregate functions denir

- **count** (eleman sayısı)
- **sum** (toplama işlemi)
- **avg** (average) ortalama
- **max** (maximum)
- **min** (minimum)

SQL: avg

| 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 |

select avg (ortalama) from öğrenci ;

| | |
|---|-------------------------|
| | avg double precision |
| 1 | 3.52000002861023 |

SQL: max

| 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 |

select * from öğrenci where ortalama = (select **max (ortalama) from öğrenci) ;**

| | ogrencino integer | isim character varying | kullanıcıadı character varying | yas integer | ortalama real |
|---|----------------------|---------------------------|-----------------------------------|----------------|------------------|
| 1 | 17327 | Mustafa | m.4417 | 18 | 4 |

SQL: min

| 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 |

select * from öğrenci where ortalama= (select min(ortalama) from öğrenci) ;

| | ogrencino integer | isim character varying | kullanıcıadı character varying | yas integer | ortalama real |
|---|----------------------|---------------------------|-----------------------------------|----------------|------------------|
| 1 | 17236 | Cemil | Cm3418 | 18 | 2.9 |

SQL: Update

| 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 |

UPDATE öğrenci
SET ortalama=ortalama+1
WHERE
 ortalama =
 (select max(ortalama)
 from öğrenci) ;

| | ogrencino integer | isim character varying | kullanıcıadı character varying | yas integer | ortalama real |
|---|----------------------|---------------------------|-----------------------------------|----------------|------------------|
| 1 | 17532 | Eymen | Eymen234 | 19 | 3.2 |
| 2 | 17347 | Kemal | Kml45 | 17 | 3.7 |
| 3 | 17236 | Cemil | Cm3418 | 18 | 2.9 |
| 4 | 16458 | Hayri | Hayri95 | 19 | 3.8 |
| 5 | 17327 | Mustafa | m.4417 | 18 | 5 |

SQL: delete

| 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 |

X

DELETE from öğrenci where ortalama= (select min (ortalama) from öğrenci) ;