Veritabanı Yönetim Sistemleri (335)

Yrd. Doç. Dr. Ahmet Arif AYDIN

October 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 ≡ ifade2
- Aynı
  - ifade1 ve ifade2 birebir aynı ise (identically equal) ikisinin manaları da aynıdır

- Yeniden yazma kuralları
  - Bir ifadede ifade1 kullanılmılsa aynı ifade ifade2 ile de tekrar yazılabilir.

İmplication Law

Implication : gerektirmek, ima etmek

- P ➔ Q
  - If p then q ≡ **(not p) or q**
  - Eğer p dogru ise q da dogrudur (tersi doğru değildir)

  - P = Hava kapalıdır
  - Q = Güneş görünmemektedir

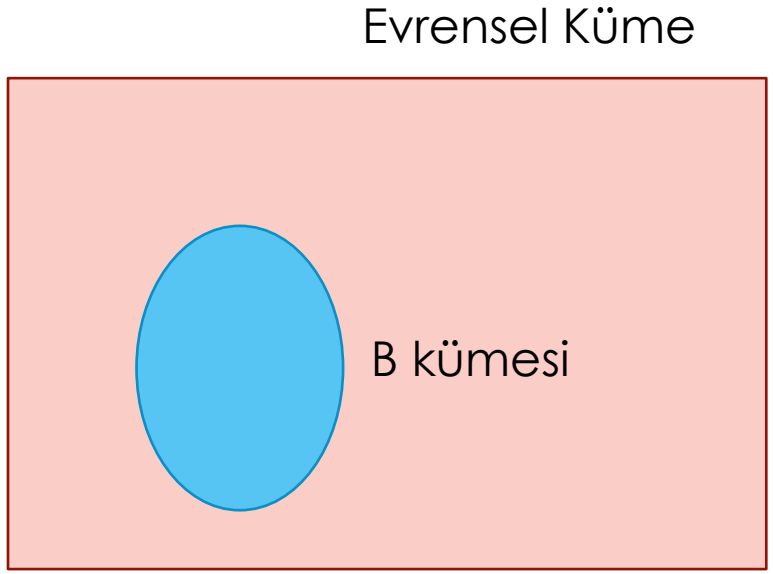
P ➔ Q **eşit degildir** Q ➔ P

http://whatis.techtarget.com/definition/logical-implication

Double Negation Law

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

- not (not p) ≡ p



De Morgan Kuralı

DeMorgan's Law

- not (p and q) ≡ (not p) or (not q)
- not (p or q) ≡ (not p) and (not q)

Distributive Law

Dağıtım Kuralı

- p and (q or r) ≡ (p and q) or (p and r)
- p or (q and r) ≡ (p or q) and (p or r)

İlişkisel Hesap (Relational Calculus)

- İlişkisel cebirin (relational algebra) bir alternatifidir.
- Declerativedir.
- Cevapları nasıl hesaplanacağının detayları verilmez (nonprocedural)
- SQL dilinin ortaya cıkmasında 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 | **P(t)** } veya {t | **şart(t)** } t = **şartı sağlayan tuple'lar** (tabloda bulunan satırlar)

- { t | **işçi(t)** and t.yaş > 40}
  - Yaşı 40 dan büyük olan işçilerin hepsinin bütün bilgilerini (tuple) getir

- { t | **personel (t)** and t.bölümno = 10}
  - 10 nolu bölümde çalışan personelin hepsinin bütün bilgilerini (tuple) getir

Tuple Relational Calculus (TRC)

- { t | **şart(t)** }
- Şart:
  - Atomik formül
  - **¬ P =** not P (p nin değili)
  - P Λ Q = P ve Q
  - (P V Q) = P veya Q
  - P ➔ Q = P doğru ise Q da doğrudur
  - ∃ R (şart (R)) : (∃: en az bir tane var(**there exists**))
  - ∀ K (şart (K)) (∀: hepsi için (for all))

Domain Relational Calculus

Domain Relational Calculus (DRC)

- { < sütun1, sütun2, sütun3, … sütunn > | P (sütun1, sütun2, sütun3, … sütunn) }
- Hesaplanan sonuc: **tuple(satır) yerine seçilen sutunlardır.**

- **{<işçi.no, işçi.adı > | < işçi.no, işçi.adı > ? Işçi ∧ bölüm.no = 10}**
  - 10 nolu bölümde çalışan işçilerin adını ve işçi nosunu getir.

https://www.tutorialcup.com/dbms/relational-calculus.htm

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 | şart(t) }

{ *T* | K **E** KullanıcıReyting /\ *K. 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 reytıngı veren kullanıcıno alanlarını getir.

{ t | şart(t) }

{ T | ∃ K ∈ KullanıcıReyting

(K.reyting > 5 /\

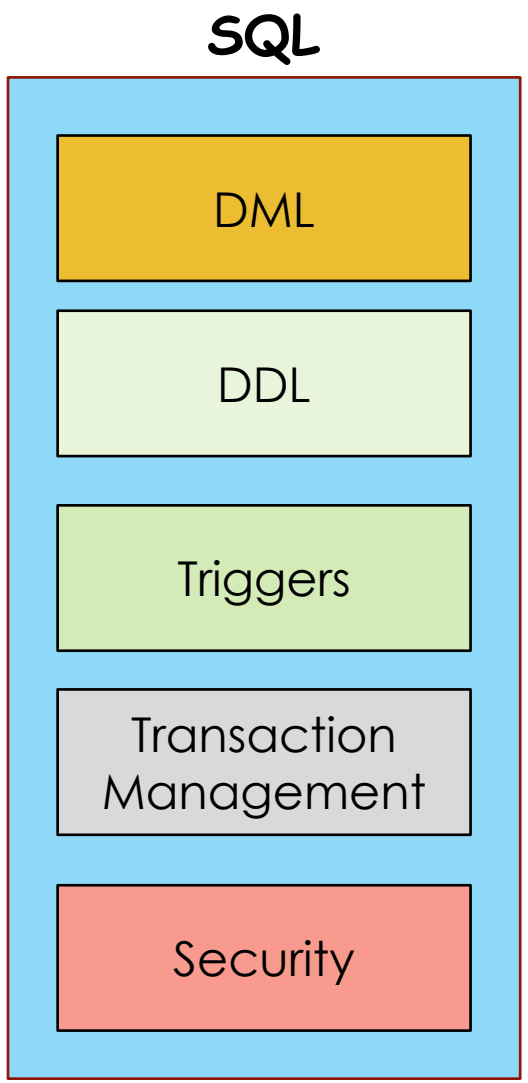
Tkullanıcı =K.kullanıcıno /\

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



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

Data Manipulatıon 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 nesnelerine erişim denetimi sağlanır.

SQL: Delete Table

ögrenci

ögrencino	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

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

Bir tabloyu silmek için

```
DROP TABLE ögrenci;
```

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

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

Char (n)
Ayrılan alan sabit

Varchar (n),
Ayrılan alan değişken

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

SQL: insert into

ögrencino	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

Tablonun bütün kolonlarına ekleme yapar

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

2- **insert into** ögrenci (ögrencino, isim, kullaniciadi, yas, ortalama) **values** (17532, 'Eymen', 'Eymen234',19,3.2);

Tablonun belirtilen kolonlarına ekleme yapar

SQL: Select

ögrencino	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  ögrenci;
```

SQL: Select

ögrencino	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 ögrencino, isim from ögrenci;
```

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

SQL: select

ögrencino	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  ögrenci 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

ögrencino	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 ögrenci 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

ögrencino	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 ögrenci 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

ögrencino	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 ögrenci order by yas;
```

	yas integer
1	17
2	18
3	19

SQL: Aggregate functions

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

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

SQL: avg

ögrencino	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 ögrenci ;
```

	avg double precision
1	3.52000002861023

SQL: max

ögrencino	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 ögrenci where ortalama = (select max (ortalama) from ögrenci) ;
```

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

SQL: min

ögrencino	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 ögrenci where ortalama= (select min(ortalama) from ögrenci) ;
```

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

SQL: Update

ögrencino	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 ögrenci
SET ortalama=ortalama+1
WHERE
ortalama =
 (select max(ortalama)
   from ögrenci) ;
```

	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

ögrencino	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 ögrenci where ortalama= (select min (ortalama) from ögrenci) ;