



YTÜ

YILDIZ TEKNİK
ÜNİVERSİTESİ

İlişkisel Model ve İlişkisel Cebir

Öğr. Gör. Dr. Yasemin Topuz

Yıldız Teknik Üniversitesi



BLM3041 – Veritabanı Yönetimi

10 Ekim 2025 | YTÜ Bilgisayar Mühendisliği

Neler konuşacağız?

- İlişkisel Model (Relational Model)
- Veritabanı Şeması
- Anahtarlar
- Şema Diyagramları
- İlişkisel Model'de Kısıtlamalar (Constraints)
- İlişkisel Cebir

İlişkisel Veritabanlarının Yapısı

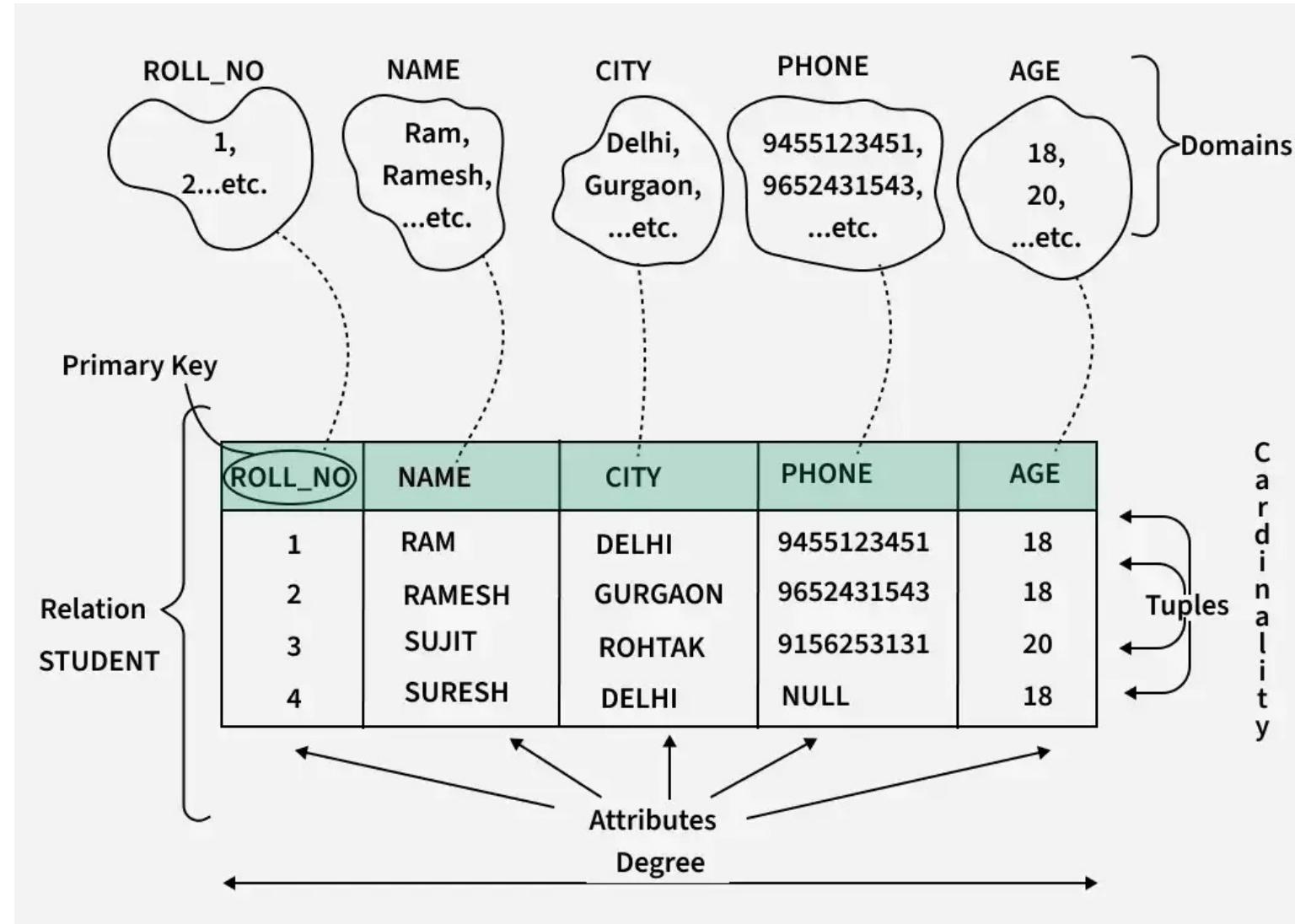
- **İlişkisel Model**, satır ve sütunlardan oluşan tabloları (ilişkileri) kullanarak verileri düzenler.
- İlişkisel model, verilerin ilişki olarak bilinen tablolar halinde düzenlendiği İlişkisel Veritabanlarında verilerin nasıl depolandığını ve yönetildiğini temsil eder.
- Tablonun her satırı bir varlığı veya kaydı temsil eder ve her sütun o varlığın belirli bir nitelğini temsil eder.

İlişkisel Veritabanlarının Yapısı

- İlişkisel Model'de, bildirim (declarative) esaslı veri işlenir.
 - Bildirim esaslı model, kullanıcıya «NE İSTEDİĞİNİ» aktarma imkanı sağlar.
- **Kavramlar**
 - **Biçimsel Model:** Entity – Relation bağıntısı üzerinden gerçekleşir ve varlıklar arasında ilişkiler vardır.
 - **Pratik Model:** SQL
 - Tablo
 - Kolon
 - Attribute
 - Satır (Tuples)
 - Domain
 - ✓ Veri Tipi
 - ✓ Fornat
 - Relational state
 - ✓ Niteliklerin Kartezyen çarpımının alt kümesi.

İlişkisel Model

Örnek: Tabloda ROLL_NO, NAME, CITY, PHONE ve AGE nitelikleri gösterilen STUDENT ilişkisini ele alalım.



İlişkisel Model – Temel Terimler

- **Nitelik (Attribute):** Nitelikler, bir varlığı tanımlayan özelliklerdir.
 - ✓ Örneğin, ROLL_NO, NAME, ADDRESS vb.
- **İlişki Şeması (Relation Schema):** Bir ilişki şeması, ilişkinin yapısını tanımlar ve ilişkinin adını nitelikleriyle birlikte temsil eder.
 - ✓ Örneğin, ÖĞRENCİ (SİPARİŞ_NO, AD, ADRES, TELEFON ve YAŞ), ÖĞRENCİ için ilişki şemasıdır.
- **Tuple:** Tuple, bir ilişkideki bir satırı temsil eder. Her tuple, belirli bir varlığı tanımlayan bir dizi öznitelik değeri içerir.
 - ✓ Örneğin, (1, RAM, DELHI, 9455123451, 18), STUDENT tablosundaki bir tuple'dır.

İlişkisel Model – Temel Terimler

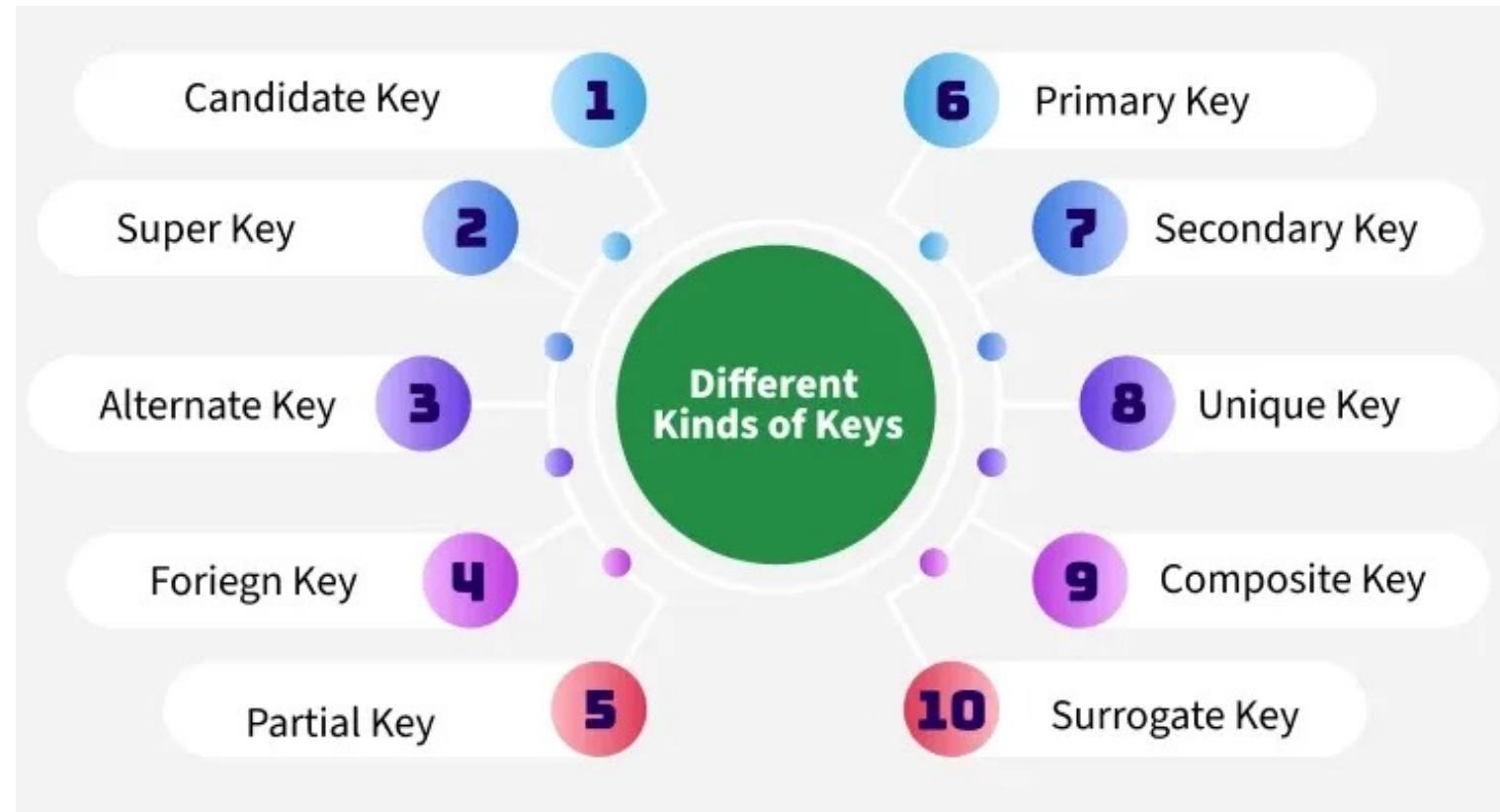
- **İlişki Örneği (Relation Instance):** Belirli bir zamanda bir ilişkinin ikili kümelerine ilişki örneği denir. Veritabanında bir ekleme, silme veya güncelleme olduğunda değişim olabilir.
- **Derece (Degree):** İlişkideki öznitelik sayısına, ilişkinin derecesi denir.
 - ✓ Örneğin, ÖĞRENCİ ilişkisi 5 öznitelijke sahip olduğundan derecesi 5'tir.
- **Kardinalite (Cardinality):** Bir ilişkideki değişkenlerin sayısına kardinalite denir .
 - ✓ Örneğin, STUDENT ilişkisinin kardinalitesi 4'tür.
- **NULL Değerleri (NULL Values):** Bilinmeyen veya erişilemeyen değerlere NULL değeri denir.
 - ✓ Örneğin, ROLL_NO 4 olan ÖĞRENCİ'nin TELEFONU NULL'dur.

İlişki Şemasının Bileşenleri

- **İlişki Adı:** Veritabanında saklanan tablonun adı. Benzersiz olmalı ve tabloda saklanan verilerle ilişkili olmalıdır. Örneğin, adı 'EMPLOYEES' olan tabloda çalışanın verileri saklanır.
- **Öznitelik Adı:** Öznitelikler, tablodaki her sütunun adını belirtir. Her özniteligin belirli bir veri türü vardır.
- **Alanlar:** Her öznitelik için olası değerler kümesi. Her sütun veya öznitelikte depolanabilecek veri türünü belirtir; örneğin tam sayı, metin veya tarih.
- **Birincil Anahtar:** Birincil anahtar, her bir değişkeni benzersiz şekilde tanımlayan anahtardır. Benzersiz olmalı ve boş olmamalıdır.
- **Yabancı Anahtar:** Yabancı anahtar, iki tabloyu birbirine bağlamak için kullanılan anahtardır. Başka bir tablonun birincil anahtarını ifade eder.
- **Kısıtlamalar:** Verilerin bütünlüğünü ve geçerliliğini sağlayan kurallar. Yaygın kısıtlamalar arasında NOT NULL, UNIQUE, CHECK ve DEFAULT bulunur.

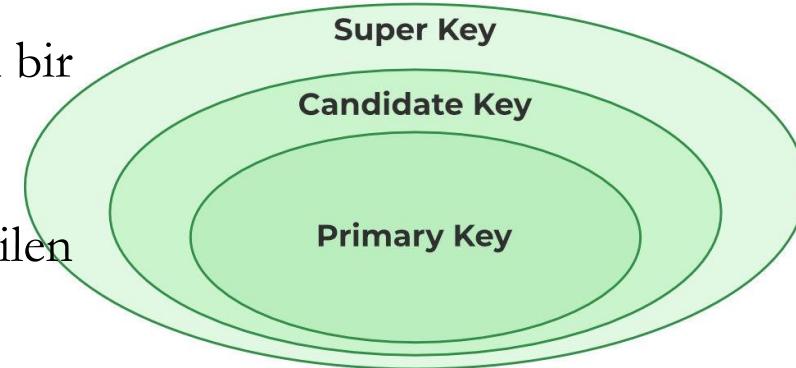
İlişkisel Model – Anahtarlar

- İlişkisel veritabanında anahtarlar, ilişkisel veritabanı modelinin temel gereksinimlerinden biridir.
- Anahtarlar, veri bütünlüğünü, benzersizliğini ve verimli erişimi sağlayan temel bileşenlerdir.
- Tablodaki satırları (tuple'ları) benzersiz şekilde tanımlamak için kullanılır.
- Ayrıca ilişkisel bir veritabanının çeşitli sütunları ve tabloları arasında ilişkiler kurmak için de anahtarları kullanırız.

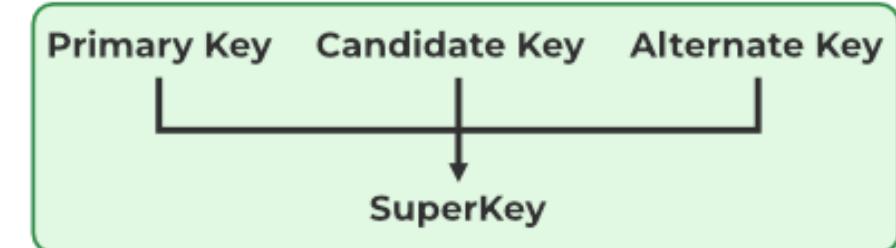
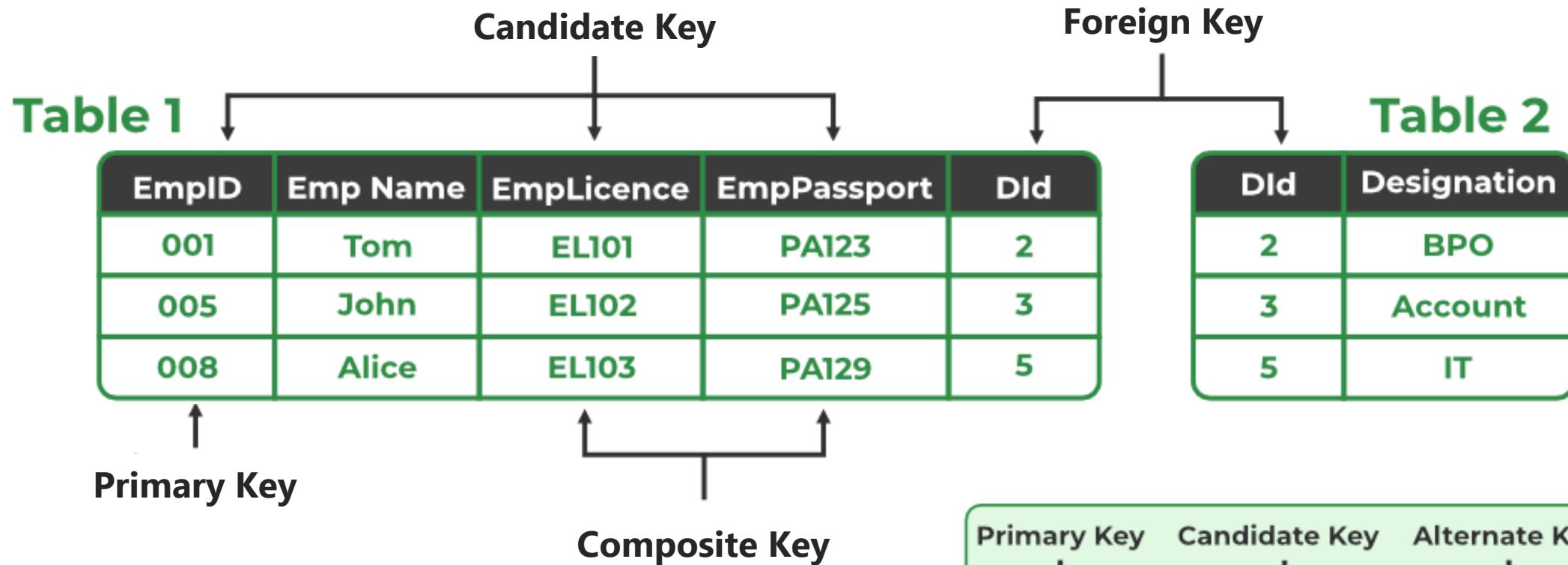


İlişkisel Model – Anahtar Türleri

- **Süper Anahtar (Super Key)**: Bir tuple'ı benzersiz şekilde tanımlayabilen bir veya daha fazla öznitelik (sütun) kümesidir.
- **Aday Anahtar (Candidate Key)**: Bir kaydı benzersiz şekilde tanımlayabilen ancak ekstra öznitelik içermeyen minimal bir süper anahtardır.
- **Birincil Anahtar (Primary Key)**: Bir ilişkideki her bir değişkeni benzersiz şekilde tanımlar. Benzersiz değerler içermelidir ve NULL değerine sahip olamaz.
- **Yabancı Anahtar (Foreign Key)**: Bir ilişkideki, başka bir ilişkinin birincil anahtarına atıfta bulunan bir niteliktir.
- **Bileşik Anahtar (Composite Key)**: Bir tuple'ı benzersiz şekilde tanımlamak için iki veya daha fazla özniteligin birleştirilmesiyle oluşturulur.



İlişkisel Model – Anahtar Türleri (Örnek)



<https://www.geeksforgeeks.org/dbms>

İlişkisel Model Notasyonu

- Verilen $R(A_1, A_2, \dots, A_n)$ ilişkisinde;
 - $r(R) \subset \text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n)$; $r(R)$: Relation State
 - $R(A_1, A_2, \dots, A_n)$ ise ilişkinin şemasıdır.
 - $r(R)$, R ilişkisinin spesifik bir state'ıdır ("value", "population") ; satırların (tuples) kümesidir.
 - $r(R) = \{t_1, t_2, \dots, t_n\}$ n satırın kümesidir.
 - $t_i = \langle v_1, v_2, \dots, v_n \rangle$ ve her $v_j \in \text{dom}(A_j)$ 'nin bir elementidir.
- $R(A_1, A_2)$ bir ilişkisel şema olsun:
 - $\text{dom}(A_1) = \{0, 1\}$, $\text{dom}(A_2) = \{a, b, c\}$
 - $\text{dom}(A_1) \times \text{dom}(A_2)$ ifadesinin olası kombinasyonları:
 - $\{ \langle 0, a \rangle, \langle 0, b \rangle, \langle 0, c \rangle, \langle 1, a \rangle, \langle 1, b \rangle, \langle 1, c \rangle \}$
 - Örnek: relation state $r(R) \subset \text{dom}(A_1) \times \text{dom}(A_2)$.
 - R ilişkisinin A_1 ve A_2 üzerinden tanımlanan olası bir r state'ıdır.
 - 3 adet 2-tuple'dan oluşur: $\{ \langle 0, a \rangle, \langle 0, b \rangle, \langle 1, c \rangle \}$ olabilir.
- Tüm değerler atomic (indivisible) olarak değerlendirilir.
- Bilinmeyen değerler için NULL değeri kullanılır.
- $S = \{R_1, R_2, \dots, R_n\}$
 - S : Tüm veritabanı; R_1, R_2, \dots, R_n ise bireysel ilişki şemalarıdır.
- Gösterim (Notation):
 - t tuple'ının değerleri **İNÇİN** $t[A]$ veya $t.A_i$ gösterimi kullanılır.

İlişkisel Model Notasyonu

Informal Terms (Pratik model)	Formal Terms (Biçimsel model)
Table	Relation
Column Header	Attribute
All possible Column Values	Domain
Row	Tuple
Table Definition	Schema of a Relation
Populated Table	State of the Relation

Schema tablo yapısını, state ise o anki satır kümesini ifade eder.

İlişkisel Modelin Özellikleri

- **Veri Temsili:** Veriler tablolarda (ilişkilerde) düzenlenir; satırlar (demetler) kayıtları, sütunlar (öznitelikler) ise veri alanlarını temsil eder.
- **Atomik Değerler:** Tablodaki her öznitelik atomik değerler içerir; bu, tek bir hücrede çok değerli veya iç içe geçmiş verilere izin verilmemiş anlamına gelir.
- **Benzersiz Anahtarlar:** Her tablonun, her kaydı benzersiz şekilde tanımlayan birincil bir anahtarı vardır ve bu sayede yinelenen satırlar oluşmaz.
- **Öznitelik Alanı:** Her özniteligin, tutabileceği değerler için geçerli veri türlerini ve kısıtlamaları belirten tanımlanmış bir alanı vardır.
- **Veri Bağımsızlığı:** Model, mantıksal ve fiziksel veri bağımsızlığını garanti altına alarak, uygulama katmanını etkilemeden veritabanı şemasında değişiklik yapılmasına olanak tanır.
- **İlişkisel İşlemler:** Seçim, yansıtma, birleştirme, birleştirme ve kesişim gibi işlemleri destekleyerek güçlü veri alma manipülasyonuna olanak tanır.
- **Veri Tutarlılığı:** Kısıtlamalar yoluyla veri tutarlığını sağlar, yedekliliği ve anormallikleri azaltır.
- **Küme Tabanlı Gösterim:** İlişkisel modeldeki tablolar kümeler olarak ele alınır ve işlemler matematiksel kümeye teorisi prensiplerini takip eder.

İlişkisel Modeldeki Kısıtlamalar (Constraints)

- İlişkisel Modeli tasarlarken, veritabanındaki veriler için geçerli olması gereken bazı koşulları tanımlarız. Bunlara Kısıtlamalar denir.
- Bu kısıtlamalar, veritabanında herhangi bir işlem (ekleme, silme ve güncelleme) gerçekleştirilmeden önce kontrol edilir.
- Kısıtlamalardan herhangi biri ihlal edilirse, işlem başarısız olur.

İlişkisel Modeldeki Kısıtlamalar (Constraints)

- **Kısıtlamalar**, tüm geçerli ilişki durumlarında tutulması gereken koşullardır.
- 3 ana kısıt tipi vardır:
 - **Key Constraints** (Anahtar Kısıtı)
 - **Entity Integrity Constraints** (Varlık Bütünlük Kısıtı)
 - **Referential Integrity Constraints** (İma Bütünlük Kısıtı)
- **Domain Constraint**
- **Semantic Constraints**

İlişkisel Modeldeki Kısıtlamalar (Constraints)

Key Constraints

- Veritabanındaki her ilişki, bir tuple'i benzersiz şekilde tanımlayan en az bir öznitelik kümesine sahip olmalıdır. Bu öznitelik kümesine anahtar denir.
- Örneğin; STUDENT'taki ROLL_NO, anahtardır. İki öğrenci aynı sayı numarasına sahip olamaz. Dolayısıyla bir anahtarın iki özelliği vardır:
 - Tüm tuple'lar için benzersiz olmalıdır.
 - NULL değerleri olamaz.

İlişkisel Modeldeki Kısıtlamalar (Constraints)

Entity Integrity Constraints

- Primary Key herhangi bir satıda **NULL** değerine sahip olamaz.
- E-posta adresi UNIQUE
 - MHRŞ'de primary key TC Kimlik No iken e-posta adresi UNIQUE olarak kısıtlanmış olsun. Sizce bu doğru bir yaklaşım mıdır?

İlişkisel Modeldeki Kısıtlamalar (Constraints)

Referential Integrity Constraints

- Bir ilişkinin bir niteliği yalnızca aynı ilişkinin veya başka herhangi bir ilişkinin başka bir niteligidenden değer alabildiğinde, buna referans bütünlüğü denir.
- İki ilişkimiz olduğunu varsayıalım
- STUDENT'ın BRANCH_CODE'u, yalnızca BRANCH'in BRANCH_CODE'unda bulunan değerleri alabilir ya da NULL olabilir; bu, referans bütünlüğü kısıtlaması olarak adlandırılır.

Table: STUDENT

ROLL_NO	NAME	ADDRESS	PHONE	AGE	BRANCH_CODE
1	RAM	DELHI	9455123451	18	CS
2	RAMESH	GURGAON	9652431543	18	CS
3	SUJIT	ROHTAK	9156253131	20	ECE
4	SURESH	DELHI		18	IT

Table: BRANCH

BRANCH_CODE	BRANCH_NAME
CS	COMPUTER SCIENCE
IT	INFORMATION TECHNOLOGY
ECE	ELECTRONICS AND COMMUNICATION ENGINEERING
CV	CIVIL ENGINEERING

İlişkisel Modeldeki Kısıtlamalar (Constraints)

Domain Constraints

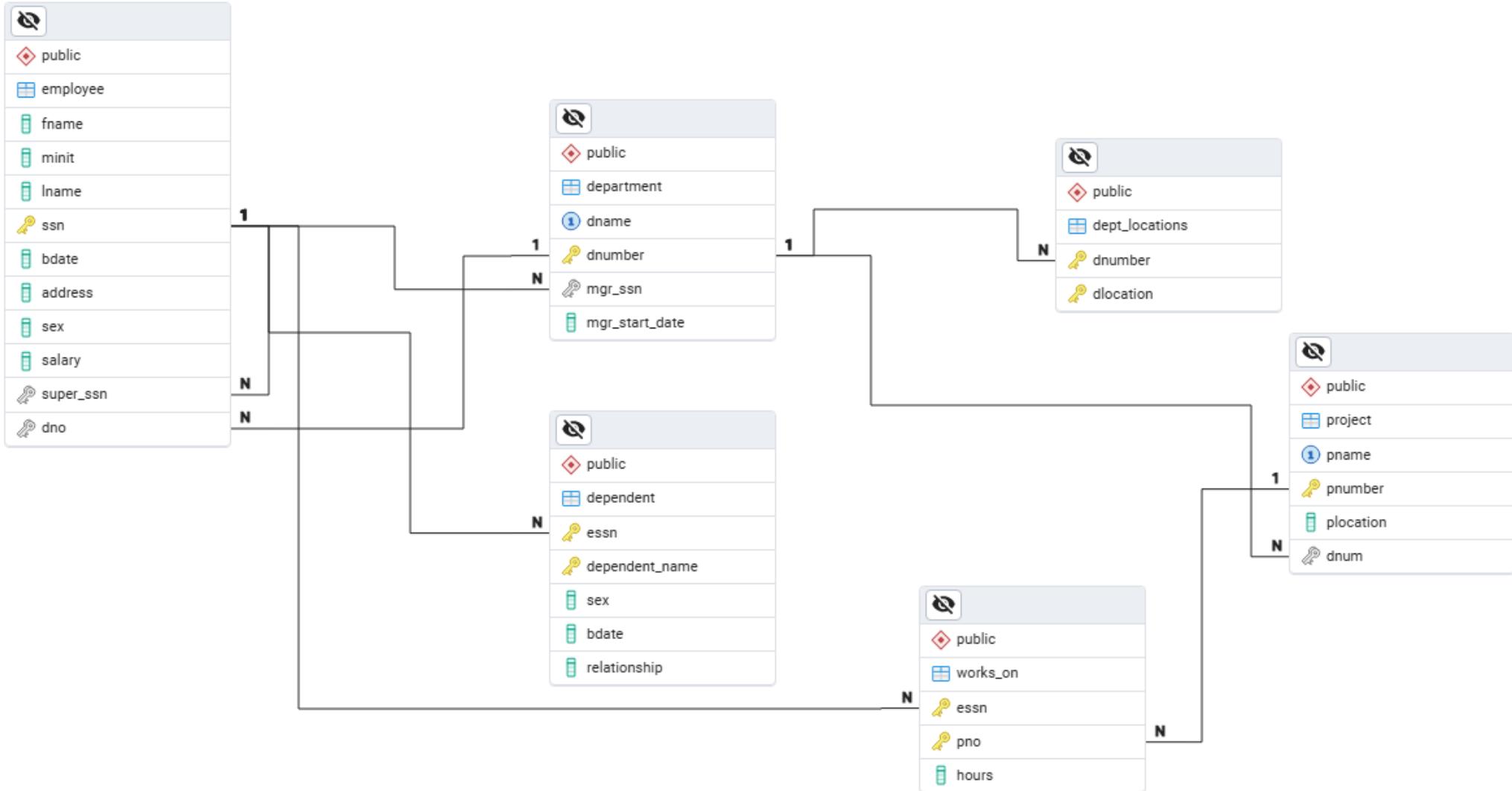
- Alan Kısıtlamaları, bir tuple içindeki her A öznitelığının değerinin, belirtilen etki alanı $\text{dom}(A)$ 'dan türetilen atomik bir değer olmasını sağlar. Alanlar, özniteliklerle ilişkili veri türleri tarafından tanımlanır.
Yaygın veri türleri sunlardır:
- **Sayısal türler (Numeric types):** Tam sayılar için tam sayıları (kısa, normal ve uzun) ve ondalık değerler için gerçek sayıları (kayan noktalı sayı, çift duyarlıklı) içerir, böylece hassas hesaplamalar yapılabilir.
- **Karakter türleri (Character types):** Çeşitli boyutlardaki metin verilerini depolamak için sabit uzunluktaki (CHAR) ve değişken uzunluktaki (VARCHAR, TEXT) dizelerden oluşur.
- **Boolean değerleri (Boolean values):** Doğru veya yanlış değerlerini depolar, genellikle veritabanlarında bayraklar veya koşullu kontroller için kullanılır.
- **Özelleştirilmiş türler (Specialized types):** Zamanla ilgili ve finansal verilerin hassas bir şekilde işlenmesi için kullanılan tarih (DATE), saat (TIME), zaman damgası (TIMESTAMP) ve para (MONEY) türlerini içerir.

İlişkisel Modeldeki Kısıtlamalar (Constraints)

Semantic Integrity Constraints

- Örnek: «Bir projede çalışanların haftalık çalışma saatleri maksimum 56 olabilir.»
- **TRIGGERS** (Tetikleyiciler)
- Integrity Violation
 - Operasyonu İptal Et
 - Operasyonu Sürdür ve Yöneticiyi Bilgilendir
 - Çeşitli Güncellemeler Yap
 - Hata Kontrol Rutinini Çağır

COMPANY DB Relational Schema Diagram



İlişkisel Model - Örnek

ENROLL tablosunda neden (**ssn**, **courseno**, **quarter**) bileşik anahtar seçildi?

Aynı öğrenci aynı dersi **farklı çeyreklerde** tekrar alabilir; ama **aynı çeyrekte** aynı dersi birden fazla kayıt satırıyla alamaz. Bu yüzden üçlü bileşik anahtar tekrarları engeller.

COURSE

	courseno [PK] text	cname text	dept text
1	CS101	Intro to Programming	CS
2	CS245	Databases	CS
3	EE210	Circuits I	EE
4	MATH201	Discrete Math	MATH

STUDENT

	ssn [PK] character (9)	name text	major text	bdate date
1	111223333	Ayşe Yılmaz	CS	2004-03-12
2	222334444	Berk Demir	EE	2003-11-02
3	333445555	Ceren Acar	CS	2004-07-19
4	444556666	Deniz Korkmaz	Math	2002-12-30
5	555667777	Efe Özkan	Bio	2003-05-05

ENROLL

	ssn [PK] character (9)	courseno [PK] text	quarter [PK] text	grade character (2)
1	111223333	CS101	2025Q1	A
2	111223333	CS245	2025Q1	B
3	333445555	CS245	2025Q1	A
4	222334444	EE210	2025Q1	B
5	444556666	MATH201	2025Q1	A
6	555667777	CS101	2025Q1	C

İlişkisel Cebir (Relational Algebra – RA)

- SQL endüstriyel standartken, RA'nın herhangi bir standardı yok!
- SQL: Sonuç odaklı tanımlayıcı bir sorgulama dili
- RA: Görev odaklı prosedürel bir sorgulama dili
- VTYS'ler SQL sorgusunu çalıştırmak için RA'ya dönüştürür
- RA bir sorgu ağacı olarak ifade edilebilir.
 - ✓ Sorgu içerisindeki tablolar ve operatörler için çalışma sırası belirleme

İlişkisel Cebir - Operatörler

- SELECT: σ
- Project: Π
- Sort: S
- Rename: P
- Extend: E
- Aggregate: \mathcal{F}
- Groupby: \mathcal{F}

İlişkisel Cebir – Operatörler (İkili Tablo)

- Union: \cup
- Intersection: \cap
- Set Difference: $-$
- Division: \div
- Product: \times
- Theta-join: \bowtie
- Natural-join: $*$ veya \bowtie
- Semijoin: \bowtie_L
- Antijoin: \triangleright
- Full Outer Join: \bowtie_{F}
- Left Outer Join: \bowtie_L
- Right Outer Join: \bowtie_R

İlişkisel Cebir – SELECT İşlemi

- Notasyon: $\sigma_p(r)$
 - p : Seçim kriteri
 - \wedge (and), \vee (or), \neg (not) işlemleriyle birleşmiş
- Tanım: $\sigma_p(r) = \{ t \mid t \in r \text{ and } p(t) \}$
- $\{ \langle \text{attribute} \rangle \} \text{ op } \{ \langle \text{attribute} \rangle \text{ or } \langle \text{constant} \rangle \}$
 - op: $=, \neq, >, \geq, <, \leq$

İlişkisel Cebir – SELECT İşlemi

A	B	C	D
α	α	1	7
α	β	5	7
β	β	12	3
β	β	23	10

$$\sigma_{A=B \wedge D > 5}^{(r)}$$

A	B	C	D
α	α	1	7
β	β	23	10

İlişkisel Cebir – Project İşlemi

- Notasyon: $\Pi_{\{A_1, A_2, \dots, A_n\}}(r)$
 - A_1, A_2, \dots : Öznitelik (kolon) adları
 - r : İlişki adı
- Sonuç, listelenmemiş kolonlar atıldıktan sonra elde edilir.
- İlişkilerde yinelenen satırlar (duplicate'lar) silinir.

A	B	C
α	10	1
α	20	1
β	30	1
β	40	2

$$\Pi_{A,C}(r)$$

A	C
α	1
β	1
β	2

İlişkisel Cebir – Union İşlemi

- Notasyon: $r \cup s$
- Tanım: $r \cup s = \{ t \mid t \in r \text{ veya } t \in s \}$
- $r \cup s$ işleminin geçerli olması için:
 - r ve s aynı sayıda öznitelijke (arity) sahip olmalıdır.
 - Öznitelik alanları uyumlu (compatible) olmalıdır.

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

 $r \cup s$

A	B
α	1
α	2
β	1
β	3

İlişkisel Cebir – Intersect İşlemi

- Notasyon: $r \cap s$
- Tanım: $r \cap s = \{ t \mid t \in r \text{ ve } t \in s \}$
- $r \cap s$ işleminin geçerli olması için:
 - r ve s aynı sayıda öznitelijke (arity) sahip olmalıdır.
 - Öznitelik alanları uyumlu (compatible) olmalıdır.

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

$r \cap s$

A	B
α	2

İlişkisel Cebir – Difference İşlemi

- Notasyon: $r - s$
- Tanım: $r - s = \{ t \mid t \in r \text{ ve } t \notin s \}$
- $r - s$ işleminin geçerli olması için:
 - r ve s aynı sayıda öznitelijke (arity) sahip olmalıdır.
 - Öznitelik alanları uyumlu (compatible) olmalıdır.

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

r - S

A	B
α	1
β	1

İlişkisel Cebir – Renaming İşlemi

- Notasyon: $P_{\text{OldName} \rightarrow \text{NewName}}(r)$

Paternity

Father	Child
Adam	Cain
Adam	Abel
Abraham	Isaac
Abraham	Ishmael

$$P_{\text{Father} \rightarrow \text{Parent}}(\text{Paternity}) \cup P_{\text{Mother} \rightarrow \text{Parent}}(\text{Maternity})$$

Maternity

Mother	Child
Eve	Cain
Eve	Seth
Sarah	Isaac
Hagar	Ishmael

Parent	Child
Adam	Cain
Adam	Abel
Abraham	Isaac
Abraham	Ishmael
Eve	Cain
Eve	Seth
Sarah	Isaac
Hagar	Ishmael

İlişkisel Cebir – Kartezyen Çarpım İşlemi

- Notasyon: $r \times s$
- Tanım: $r \times s = \{ t, q \mid t \in r \text{ ve } q \in s \}$

A	B
----------	----------

α	1
β	2

r

C	D	E
----------	----------	----------

α	10	a
β	10	a
β	20	b
γ	10	b

s

r X S

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

İlişkisel Cebir – Theta-JOIN İşlemi

- Türetilmiş bir işlemidir.
- Notasyon: $r_1 \underset{p}{\bowtie} r_2 = \sigma_p(r_1 \times r_2)$

Employees

Employee	Project
Smith	A
Black	A
Black	B

Projects

Code	Name
A	Venus
B	Mars

$\text{Employees} \bowtie_{\text{Project}} \text{Projects} = \text{Code}$

Employee	Project	Code	Name
Smith	A	A	Venus
Black	A	A	Venus
Black	B	B	Mars

İlişkisel Cebir – Natural-JOIN İşlemi

 r_1

Employee	Department
Smith	sales
Black	production
White	production

 r_2

Department	Head
production	Mori
sales	Brown

 $r_1 * r_2 \quad (r_1 \bowtie r_2)$

Employee	Department	Head
Smith	sales	Brown
Black	production	Mori
White	production	Mori

İlişkisel Cebir – Left, Right, Full Outer JOIN İşlemi

 r_1

Employee	Department
Smith	sales
Black	production
White	production

 r_2

Department	Head
production	Mori
purchasing	Brown

 $r_1 \bowtie r_2$

Employee	Department	Head
Smith	Sales	NULL
Black	production	Mori
White	production	Mori

 $r_1 \bowtie r_2$

Employee	Department	Head
Black	production	Mori
White	production	Mori
NULL	purchasing	Brown

 $r_1 \bowtie r_2$

Employee	Department	Head
Smith	Sales	NULL
Black	production	Mori
White	production	Mori
NULL	purchasing	Brown

İlişkisel Cebir – Semi-JOIN İşlemi

- Notasyon: $R \bowtie S = \Pi_{a_1, \dots, a_n} (R * S)$
 - a_1, \dots, a_n : R'nin nitelik (öznitelik) isimleri kümesidir.
- R'nin tüm nitelikleri içerisinde S'nin niteliklerinden olanlar için;
 - R'nin S ile eşit olan tüm satırlar** çekilir.

Employees

Name	Empld	Dept Name
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Production

Department

Dept Name	Manager
Sales	Bob
Sales	Thomas
Production	Katie
Production	Mark

Employees \bowtie Department

Name	Empld	DeptName
Sally	2241	Sales
Harriet	2202	Production

İlişkisel Cebir – Anti-JOIN İşlemi

- Notasyon: $R \triangleright S$
- Semi-JOIN'in tersidir.
- R'nin S ile eşit olmayan tüm satırlar** çekilir.

Employees

Name	Empld	Dept Name
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Production

Department

Dept Name	Manager
Sales	Bob
Sales	Thomas
Production	Katie
Production	Mark

Employees \triangleright Department

Name	Empld	DeptName
Harry	3415	Finance
George	3401	Finance

İlişkisel Cebir - Örnek

STUDENT (ssn, name, major, bdate)
 COURSE (Courseno, cname, dept)
 ENROLL (ssn, courseno, quarter, grade)

- Doğum tarihi 2003'ten küçük olan öğrenciler

$$\sigma_{bdate < 2003} (STUDENT)$$

- Doğum tarihi 2003'ten küçük olan ve major Math ya da Bio olan öğrenciler

$$\sigma_{bdate < 2003 \wedge (major = 'Math' \vee major = 'Bio')} (STUDENT)$$

- Doğum tarihi 2003'ten küçük olan öğrencilerin adını listele

$$\Pi_{name} (\sigma_{bdate < 2003} (STUDENT))$$



YTU

YILDIZ TEKNİK
UNİVERSİTESİ

Yasemin Topuz

Yıldız Teknik Üniversitesi



ytopuz@yildiz.edu.tr

