

Veri Yönetiminide farklı Yaklaşımlar: SQL ve NoSQL

Seyfullah Çuhadar

BEN KİMİM?

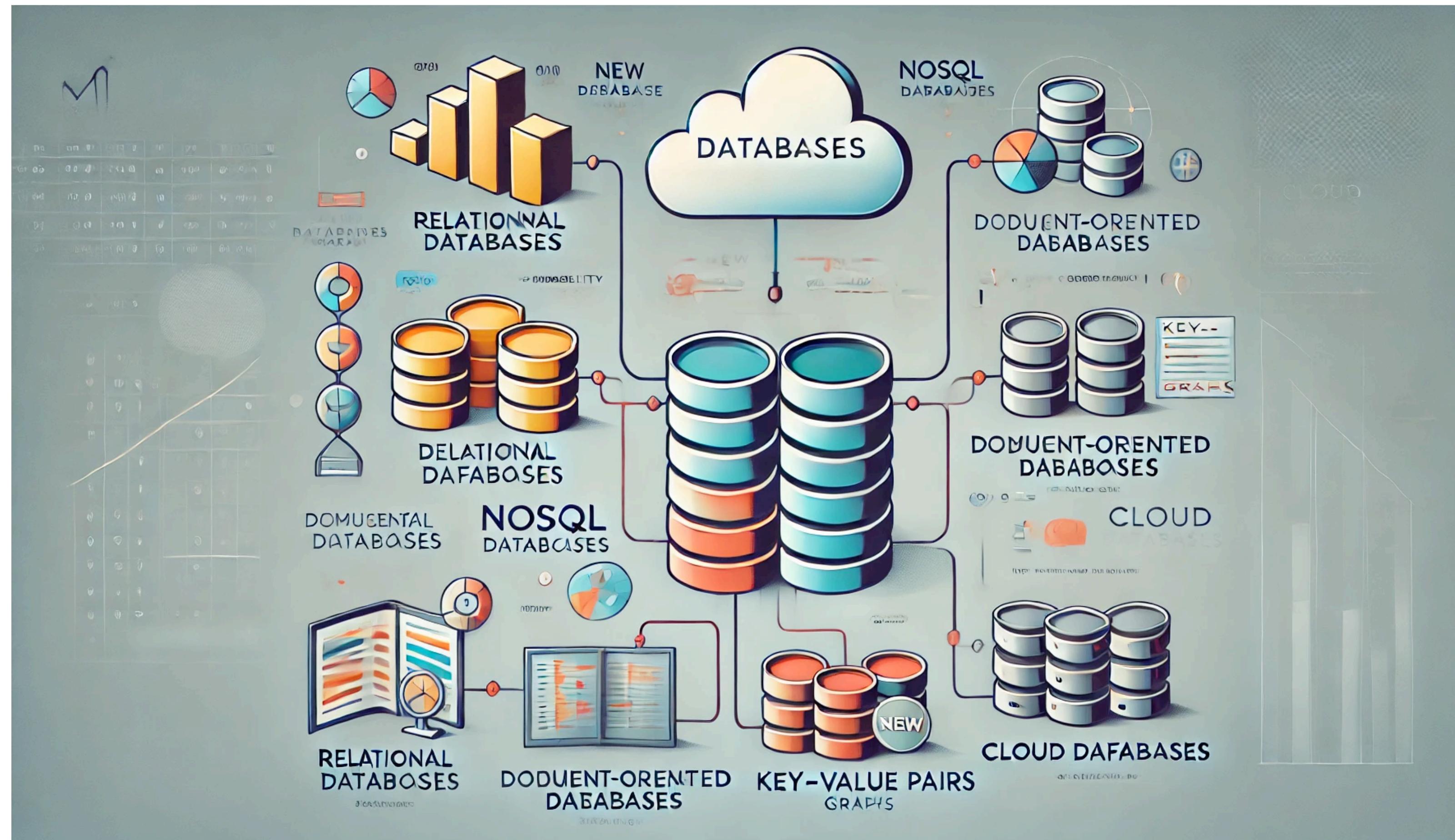
- 2017 Ankara Üniversitesi Bilgisayar Mühendisliği
- Doğum Yılı: 1994
- İş hayatımın ilk yıllarda full stack geliştirici olarak çalıştım. Şu anda back end alanında çalışıyorum
- Kullandığım teknolojiler: C# diliyle .Net çatısını kullanarak backend uygulamaları geliştiryorum
-

Veri Nedir?



Veriler tek başına anlamları olmayan, işlenmesi gereken, ham, düzensiz; sayılar, kelimeler ve nesnelerdir.

Veritabanı Nedir?



Veritabanı, verilerin düzenli bir şekilde saklandığı, erişildiği, yönetildiği ve işlendiği dijital bir depolama sistemidir.

SQL Nedir?



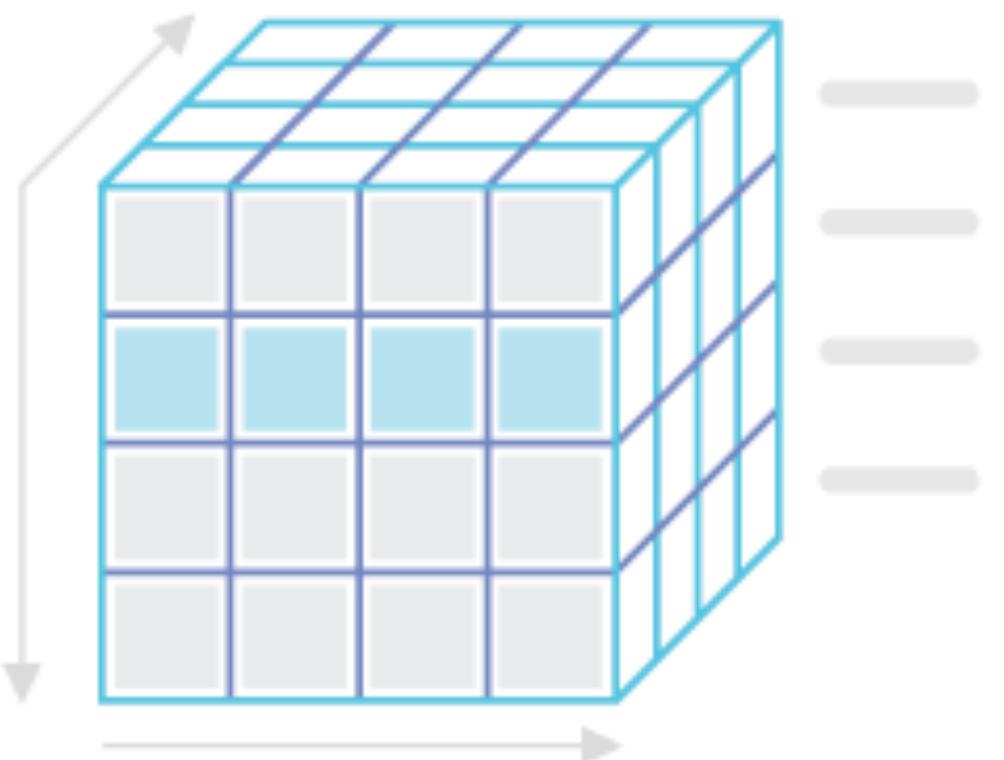
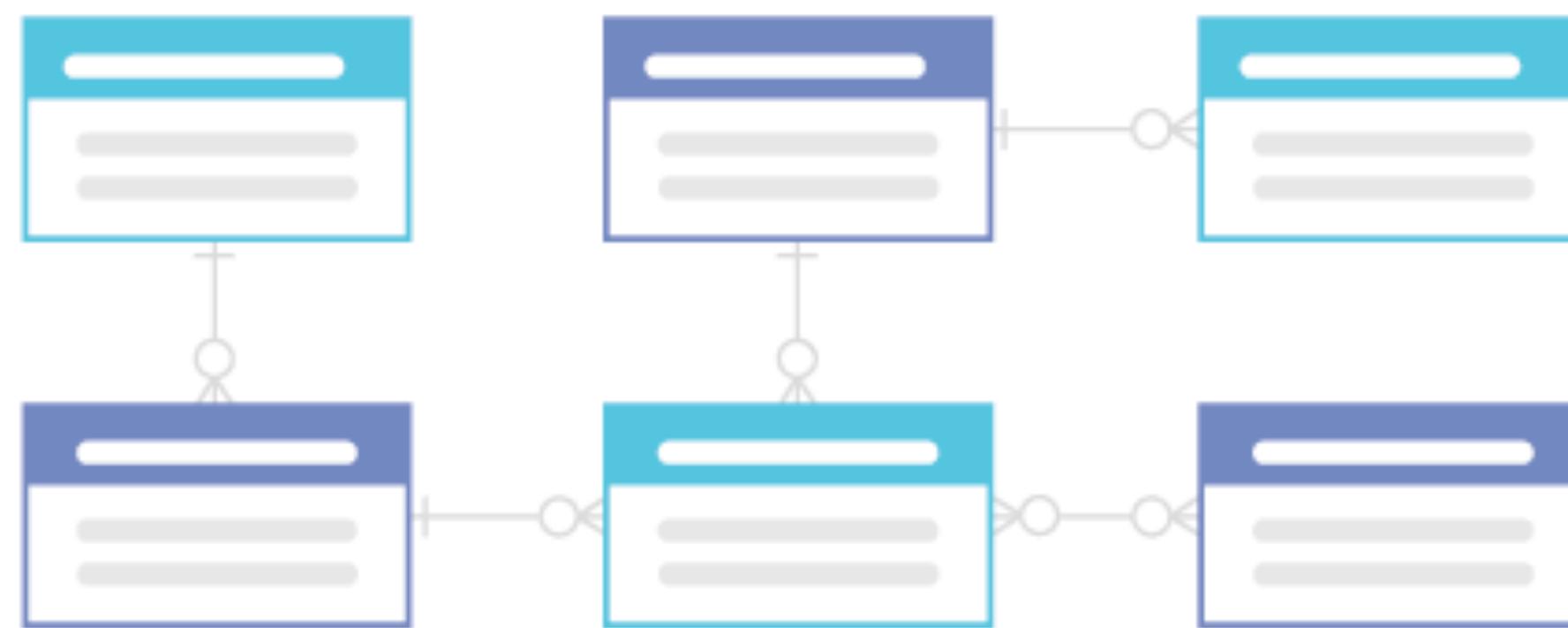
SQL bilgileri ilişkisel bir veri tabanında depolamak ve işlemek için kullanılan bir programlama dilidir.

NoSQL'in doğuşu

- 2010'larda, NoSQL, ilişkisel modelin hakimiyetini devirmeye yönelik en son girişimdir. 'NoSQL' ismi talihsiz bir seçimdir, çünkü aslında belirli bir teknolojiye atıfta bulunmaz - başlangıçta sadece 2009'da açık kaynak, dağıtık ve ilişkisel olmayan veritabanları üzerine bir buluşma için dikkat çekici bir Twitter etiketi olarak düşünülmüştü. Ancak bu terim dikkat çekti ve hızla web girişim topluluğuna ve ötesine yayıldı. Şimdi #NoSQL etiketiyle ilişkilendirilen birçok ilginç veritabanı sistemi var ve bu terim sonradan 'Sadece SQL Değil' (Not Only SQL) olarak yeniden yorumlandı

SQL

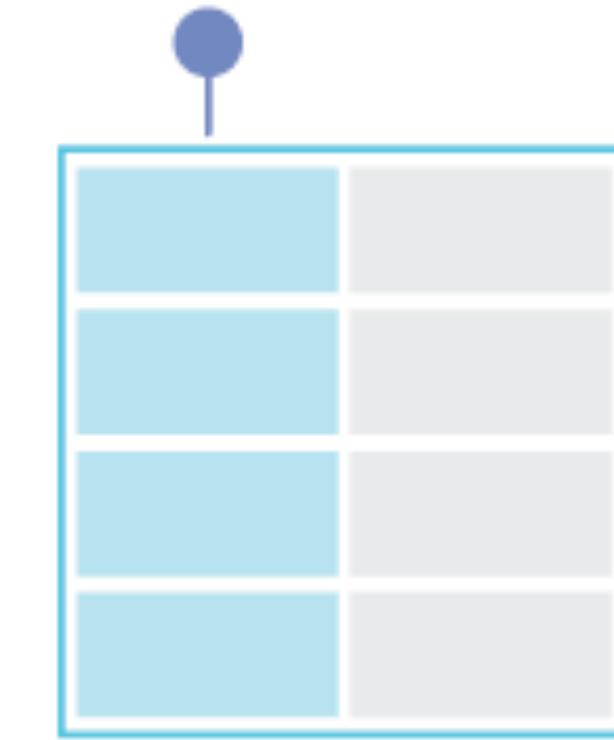
Relational Database Management Systems (RDBMS)



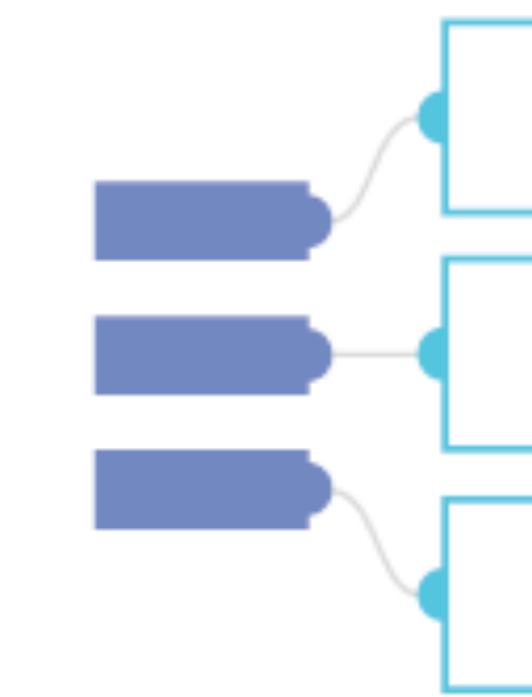
Online Analytical Processing (OLAP) Cube

NoSQL

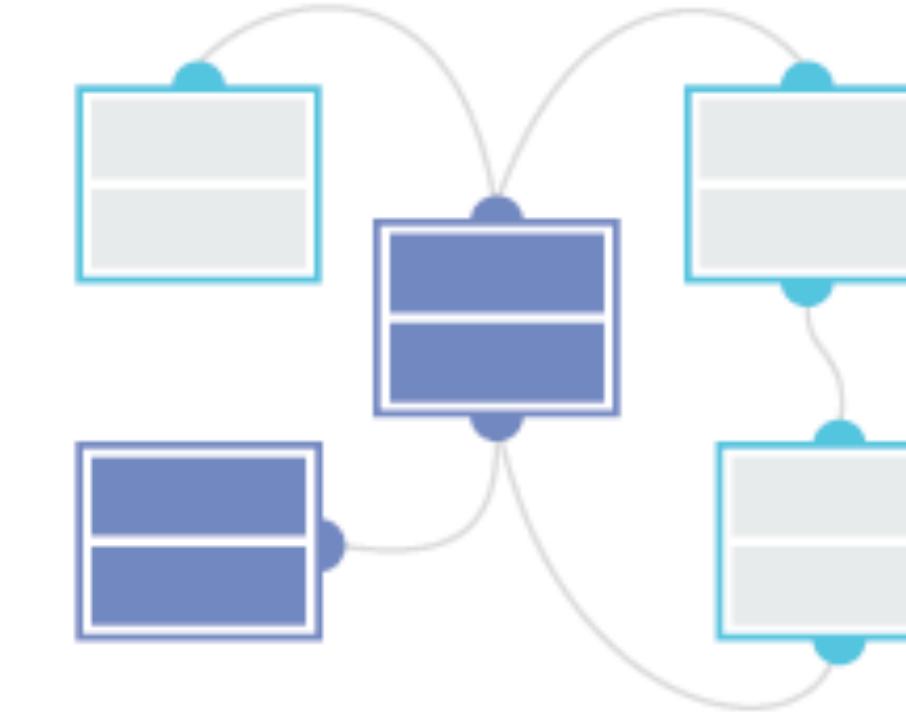
Key-Value



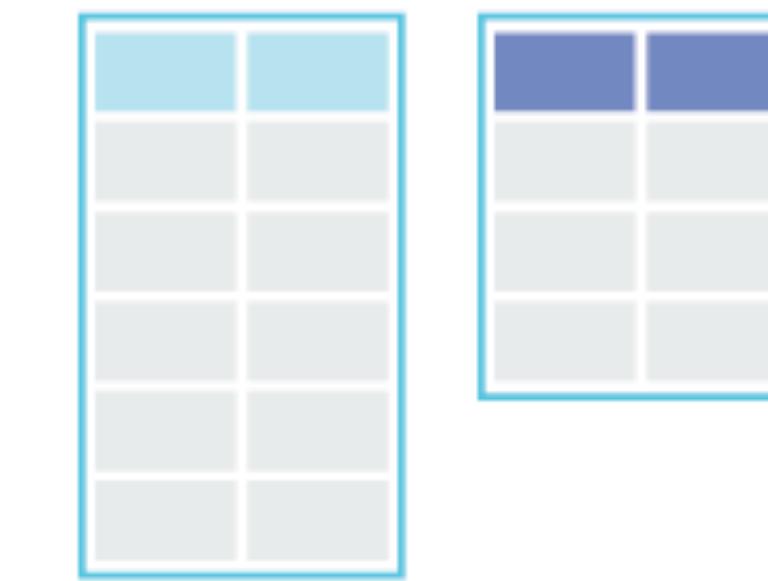
Document



Graph



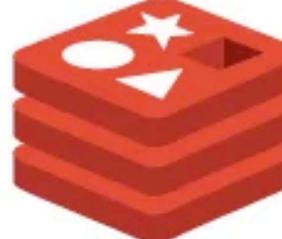
Column store



Bazı SQL Veritabanları



NOSQL Veritabanları

Type	Example	
Key-Value Store	 redis	 riak
Wide Column Store	 Apache HBase	 cassandra
Document Store	 mongoDB®	 CouchDB relax
Graph Store	 neo4j	 InfiniteGraph The Distributed Graph Database

Veri Modelleme Yöntemleri

İlişkisel model(Relational Model)

- Bugün en çok bilinen veri modeli muhtemelen, 1970 yılında Edgar Codd tarafından önerilen ilişkisel modele dayanan SQL modelidir

Veri Modelleme Yöntemleri

İlişkisel Modelin Temel Bileşenleri

- **Tablolar (İlişkiler)**: Veriler, tablolar şeklinde düzenlenir. Her tablo belirli bir varlığı temsil eder (örneğin, “Müşteriler” veya “Siparişler” gibi).
- **Satırlar (Kayıtlar veya Tuple)**: Her satır, tabloya özgü bir kayıt ya da örneği temsil eder.
- **Sütunlar (Öznitelikler)**: Tablodaki sütunlar, kayıtlardaki özellikleri veya nitelikleri belirtir.
- **Yabancı Anahtar (Foreign Key)**: Tablolar arasındaki ilişkileri temsil eder.

Transaction table

Transaction ID	Customer ID	Product ID	Purche date
53666	24221	389	06-02-2023
50333	24222	789	06-02-2023
54673	24223	879	06-02-2023
58930	24224	975	06-02-2023

Product table

Product ID	Product name	Price per kg
389	Banana	4
789	Apple	5
879	Watermelon	5
975	Mango	7

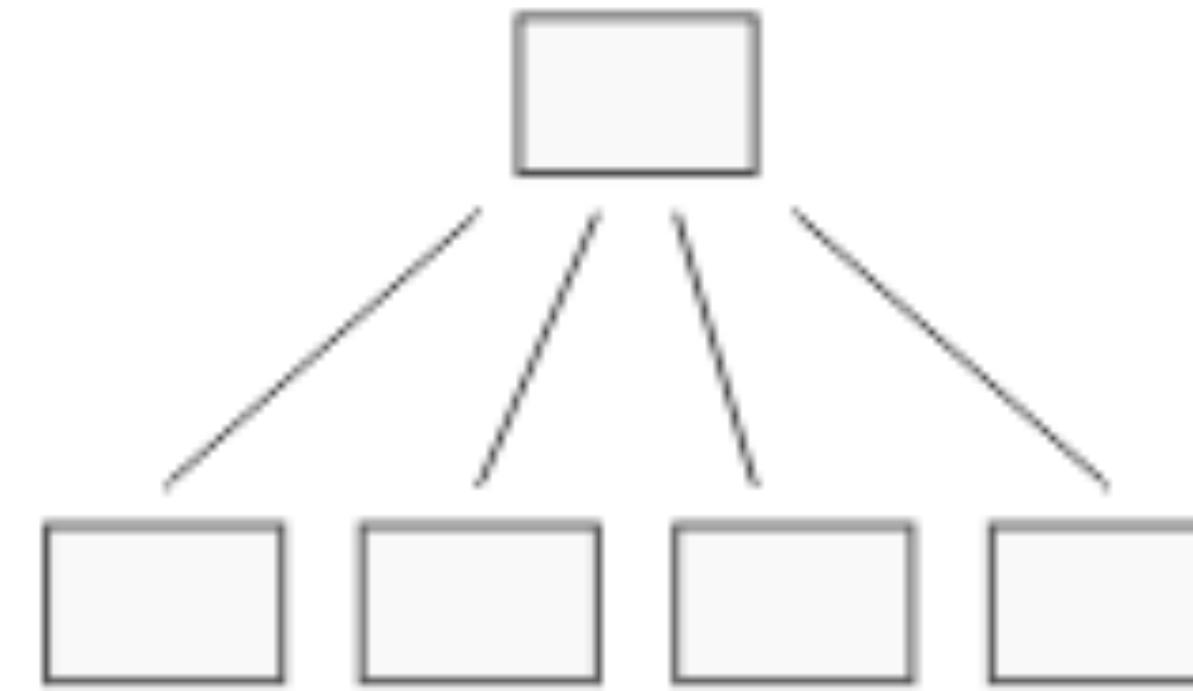
Customer table

Customer ID	Last name	First name
24221	Smith	James
24222	Jones	Sam
24223	Taylor	Ann
24224	Burton	Sue

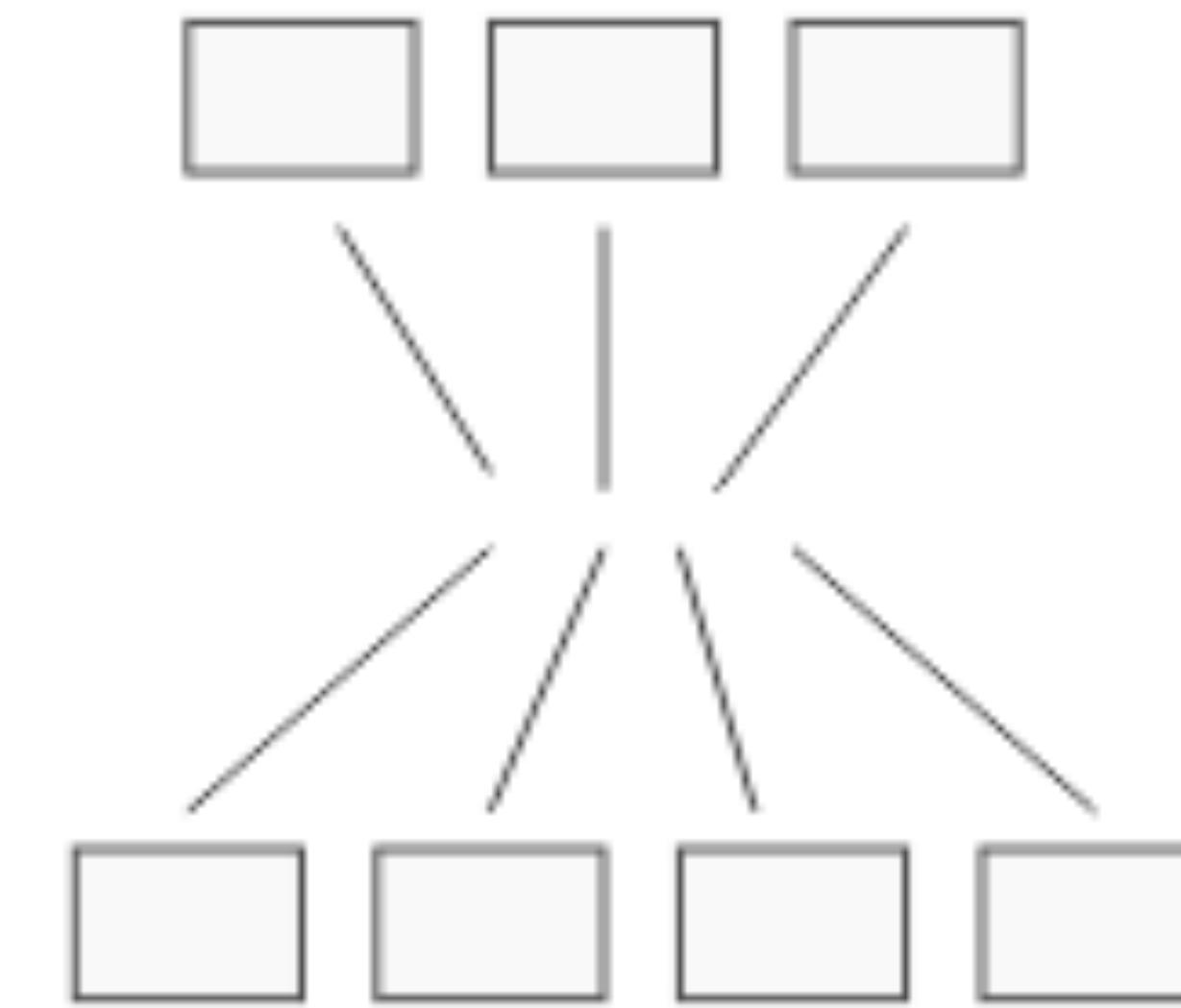
İlişki Çeşitleri(Type of relationships)



one-to-one



one-to-many



many-to-many

1-1 (Bire-bir ilişki):

ogrenci			
tc_kimlik	ad	soyad	sinif_no
111	AHMET	ATMAZ	17
222	BURAK	BURMAZ	18
333	CEYDA	CAYMAZ	18

1

kimlik_detay		
tc_kimlik	anne_kizlik_soyadi	aile_sira_no
111	YAZMAZ	17
333	DUYMAZ	21

1

1-N (Bire-çok ilişki):

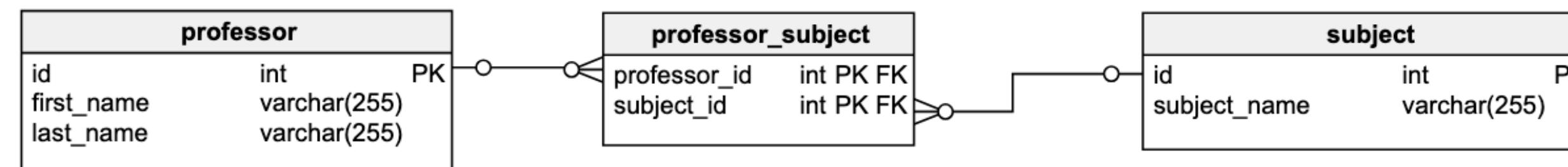
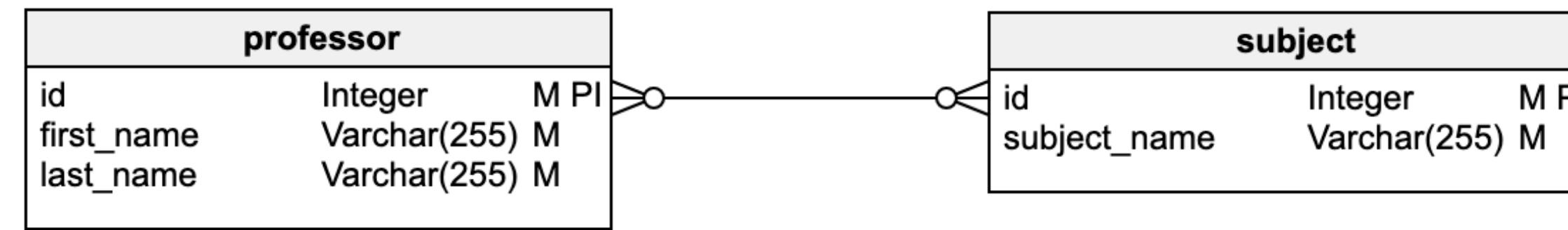
ogrenci			
<u>tc_kimlik</u>	ad	soyad	<u>sinif_no</u>
111	AHMET	ATMAZ	17
222	BURAK	BURMAZ	18
333	CEYDA	CAYMAZ	18

N

sinif		
<u>sinif_no</u>	sinif	sube
17	11	A
18	11	B

1

N-N (ÇOK-çok ilişki):





Bill Gates

Greater Seattle Area | Philanthropy

Summary

Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience

Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

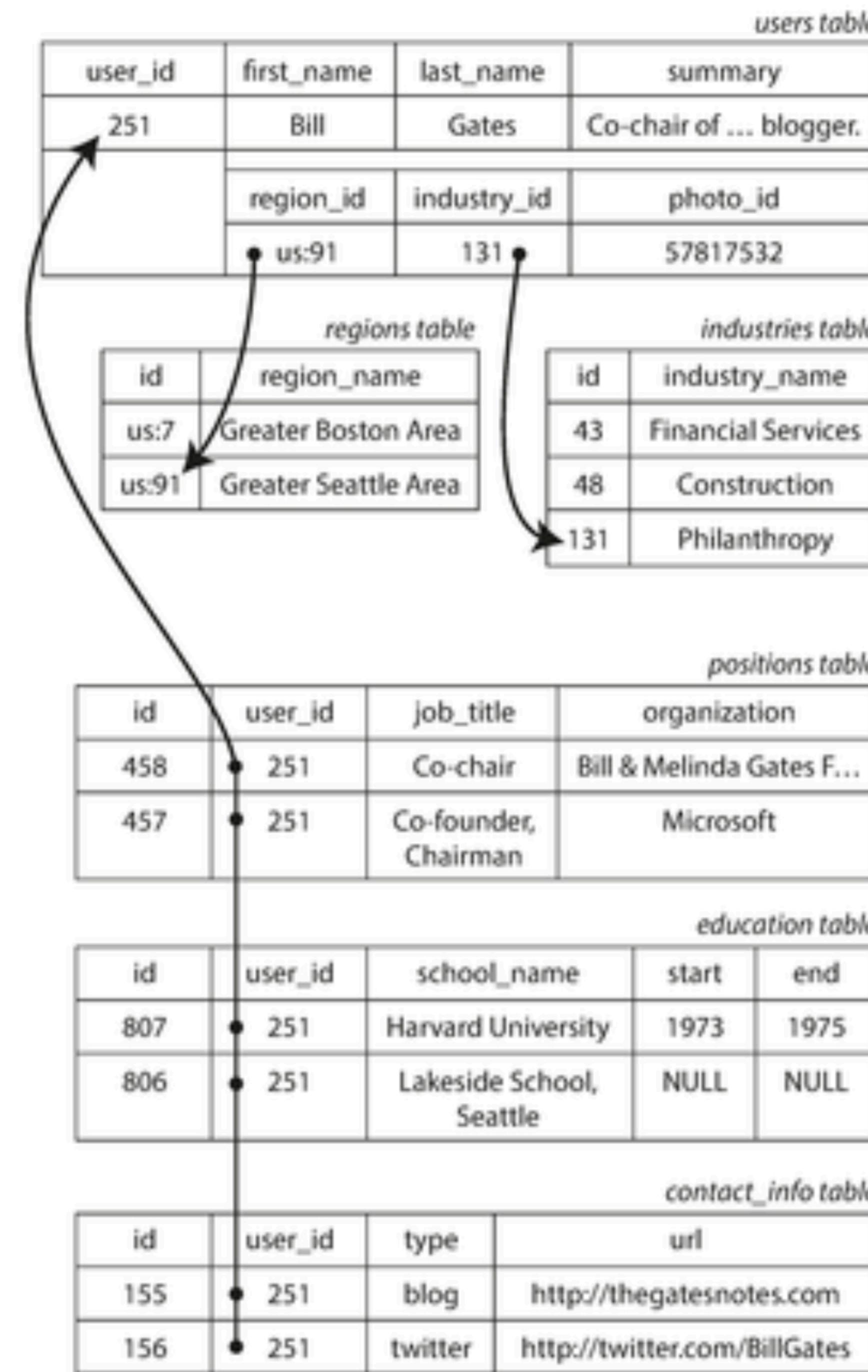
Education

Harvard University
1973 – 1975

Lakeside School, Seattle

Contact Info

Blog: thegatesnotes.com
Twitter: @BillGates



Veri Modelleme Yöntemleri

Belge Modeli(Document Model)

- verilerin belgeler şeklinde organize edildiği ve saklandığı bir NoSQL veri modeli türündür. Bu modelde, her bir belge bağımsız bir veri birimidir ve genellikle JSON, BSON (Binary JSON) veya XML gibi esnek ve hiyerarşik yapıya sahip formatlarda saklanır.

Veri Modelleme Yöntemleri

Belge Modelinin Temel Özellikleri

- Belge (Document)
- Anahtar-Değer Çifti (Key-Value Pair)
- Esnek Şema (Schema Flexibility)
- Koleksiyonlar (Collections)
- Benzersiz Kimlikler (Unique Identifiers)
- İç İçe Veri Yapıları (Nested Data Structures)
- İlişkiler (Embedded Relationships)

CUSTOMER

CUSTOMER ID

CUSTOMER NAME

CUSTOMER DATE OF BIRTH

CUSTOMER ADDRESS

ORDER DETAILS

ORDER 1

PRODUCT DETAILS

PRODUCT 1

REVIEWS

PRODUCT 2

BILLING ADDRESS

SHIPPING ADDRESS

ORDER 2

ORDER N



Bill Gates
Greater Seattle Area | Philanthropy

Summary
Co-chair of the Bill & Melinda Gates Foundation. Chairman, Microsoft Corporation. Voracious reader. Avid traveler. Active blogger.

Experience
Co-chair • Bill & Melinda Gates Foundation
2000 – Present

Co-founder, Chairman • Microsoft
1975 – Present

Education
Harvard University
1973 – 1975

Lakeside School, Seattle

Contact Info
Blog: thegatesnotes.com
Twitter: @BillGates

Example 2-1. Representing a LinkedIn profile as a JSON document

```
{  
    "user_id": 251,  
    "first_name": "Bill",  
    "last_name": "Gates",  
    "summary": "Co-chair of the Bill & Melinda Gates... Active  
blogger.",  
    "region_id": "us:91",  
    "industry_id": 131,  
    "photo_url": "/p/7/000/253/05b/308dd6e.jpg",  
    "positions": [  
        {"job_title": "Co-chair", "organization": "Bill & Melinda  
Gates Foundation"},  
        {"job_title": "Co-founder, Chairman", "organization":  
"Microsoft"}  
    "education": [  
        {"school_name": "Harvard University", "start": 1973,  
"end": 1975},  
        {"school_name": "Lakeside School, Seattle", "start": null,  
"end": null}  
    "contact_info": {  
        "blog": "http://thegatesnotes.com",  
        "twitter": "http://twitter.com/BillGates"  
}
```

Modeller arasındaki farklılıklar

Özellik	İlişkisel Model (SQL)	Belge Modeli (NoSQL)
Veri Yapısı	Tablolar ve satır-sütun yapısına dayanır	JSON, BSON veya XML formatında belgeler şeklinde
Şema	Sabit ve önceden tanımlı bir şema gerektirir	Şemasız veya esnek şema; her belge farklı yapıda olabilir
İlişki Yönetimi	Tablolar arasında ilişkiler kurulur ve JOIN işlemleri yapılır	İlişkiler genellikle tek bir belge içinde iç içe tutulur (embedded)
Veri Saklama Şekli	Veri, normalizasyon ile farklı tablolara dağıtilır	İlişkili veriler aynı belge içinde saklanabilir
Ölçeklenebilirlik	Dikeyde ölçeklenebilir (tek sunucu donanımı artırılır)	Yatayda ölçeklenebilir (yeni sunucular eklenir ve veri dağıtılır)
Veri Tutarlılığı	ACID özelliklerine dayanır; yüksek veri tutarlılığı sağlar	BASE modeline göre esnek tutarlılık sağlar
Veri Erişimi ve Sorulama	SQL sorguları ile güçlü sorulama kabiliyetine sahiptir	Basit sorulama dilleri veya API tabanlı sorgular kullanılır
Normalizasyon	Normalizasyon kurallarına göre veri tekrarını en aza indirir	Normalizasyon gerekmez; veriler tekrar edebilir
Sharding (Veri Parçalama)	Zordur, karmaşık ilişkiler nedeniyle veri parçalama işlemi sınırlıdır	Kolaydır, belgeler bağımsız olduğundan kolayca sunuculara dağıtılabılır

Nereden öğrenebilirim

- [https://youtube.com/watch?
v=hKjxTpuBZEw&list=PLjl4XZvh1FJKfsMxDeL6b8ImStVO61hhs](https://youtube.com/watch?v=hKjxTpuBZEw&list=PLjl4XZvh1FJKfsMxDeL6b8ImStVO61hhs)
- <https://learn.mongodb.com/learning-paths/introduction-to-mongodb>