



## VERİTABANI YÖNT. SİST. PRATİK NOTLAR 1 !! 😊

Celil ÖZTÜRK

19Ekim2020(Hafta 1-2)

Veri saklama birimlerinde depolanan veri topluluklarına “**dosya**” (**file**) denir.

**Neden VTYS?** Geleneksel dosya sisteminin yetersiz kaldığı durumlardaki sorunu çözmek üzere, veriyi saklama ve veriye erişim konusunda yeni yazılım teknolojilerine yönelme başlamış ve Veritabanı Yönetim Sistemleri (VTYS) yaklaşımı ortaya çıkmıştır.

Veritabanı kullanımı, geleneksel dosya kullanımına göre bir çok yönden üstünlük sağlamaktadır.

Bunlar:

- a) Verinin tekrarlamasını önler,
- b) Verinin tutarlı olmasını sağlar,
- c) Aynı andaki erişimlerde tutarsızlıkların ortaya çıkmasını önler.
- d) Verinin güvenliğini sağlar.

### KAVRAMLAR

DBMS(DATABASE MANAGEMENT SYSTEM)-Veritabanı Yönetim Sistemleri

NTP (Nesne Tabanlı Programlama)

**Veri modelleri:** Verileri, veri ilişkilerini, veri anlambilimini ve tutarlılık kısıtlamalarını açıklamak için kavramsal araçlar koleksiyonu.

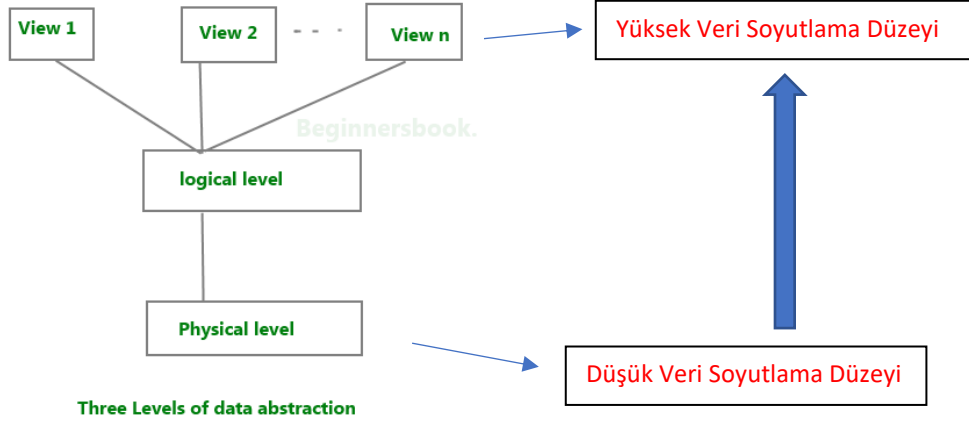
**Veri soyutlama:** Çeşitli düzeylerde veri soyutlama yoluyla veritabanındaki verileri kullanıcılardan temsil etmek için veri yapılarının karmaşıklığını gizleyin.

**Turing Machine<sub>1</sub>:** Gerçek bilgisayarların gücü, *Turing eşdeğerliği* kavramı ile belirlenir oldu; eğer bir bilgisayar, yeterli zaman ve bellek sağlandığında herhangi bir Turing makinesinin yapabileceği bir hesaplamayı yapabiliyorsa Turing eşdeğeri sayılmaktadır.

**Turing Machine<sub>2</sub>:** Turing makinesi bu programı kolayca çalıştırabilir. Ancak şimdi, herhangi bir nedenle programlama dilinizin aynı eklentiye gerçekleştiremeyeceğini hayal edin. Bu "Turing tamamlanmamış" (tabiri caizse). Öte yandan, evrensel Turing makinesinin çalıştırabileceği herhangi bir programı çalıştırabilirse, Turing tamamlandı.

**SOYUTLAMA(Data Abstraction/Veri Soyutlaması):** Veri soyutlama, veri yapıları üzerinde yapılır, kullanıcıdan alakasız ayrıntıların gizlenmesi işlemidir.

- **Fiziksel seviye:** Karmaşık veri yapısı detaylarını bu seviyede alabilirsiniz. En düşük düzey veri soyutlama.
- **Mantıksal seviye:** veritabanında depolanan verileri ve veriler arasındaki ilişkileri açıklar.
- **Görünüm düzeyi:** En yüksek düzey veri soyutlama. Veritabanı Sistemi ile kullanıcı etkileşimi sağlanan seviye !



**\*NTP/Soyutlama :** Soyutlama kavramı nesne tabanlı programlamada da detaylardan arındırma işlemi olarak karşımıza çıkar. Karmaşıklığı azaltmak için soyutlama kullanılır.

**DDL(Data Definition Language/Veri Tanımlama Dili):** Veritabanı şemaları oluşturma, tabloları, CREATE, ALTER, DROP...

**DML(Data Manipulation Language/Veri Manipulasyon Dili):** Verilere erişim ve güncelleme dili, SELECT, INSERT, UPDATE, DELETE...

**SQL(Structured Query Language/Yapısal Sorgulama Dili):**

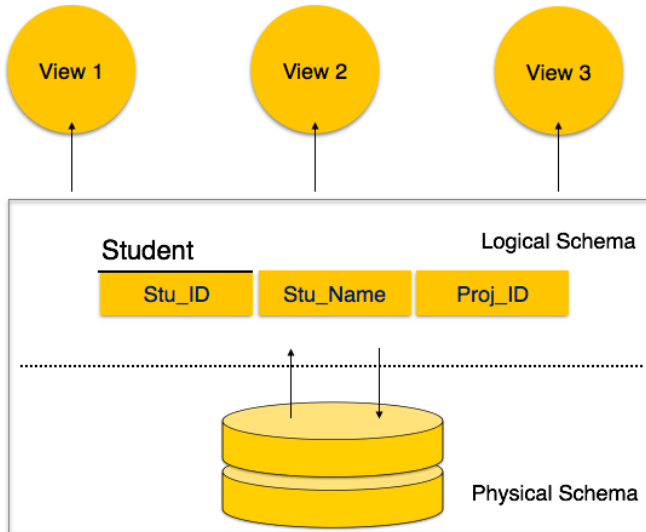
```
select name
from instructor
where dept_name = 'Comp. Sci.'
```

**DBA (DATABASE ADMINISTRATOR)-Veritabanı Yöneticisi \*:** Sistem üzerinde merkezi kontrole sahip olan kişiye veritabanı yöneticisi (DBA) denir. Bir DBA'nın işlevleri şunları içerir:

- Şema tanımı
- Depolama yapısı ve erişim yöntemi tanımı
- Şema ve fiziksel organizasyon değişikliği
- Veri erişimi için yetki verilmesi
- Rutin bakım
- Veritabanını periyodik olarak yedekleme
- Normal işlemler için yeterli boş disk alanı bulunduğundan emin olmak ve disk alanını gerektiği gibi yükseltmek
- Veritabanında çalışan işleri izleme

**Fiziksel Veritabanı Şeması** - Bu şema, verilerin gerçek depolanması ve dosyalar, indeksler, vb. Gibi depolanma biçimleriyle ilgilidir. Verilerin ikincil bir depoda nasıl depolanacağını tanımlar.

**Mantıksal Veritabanı Şeması** - Bu şema, depolanan verilere uygulanması gereken tüm mantıksal kısıtlamaları tanımlar. Tabloları, görünümüleri ve bütünlük kısıtlamalarını tanımlar.



### VARLIK-İLİŞKİ MODELİ(E-R)

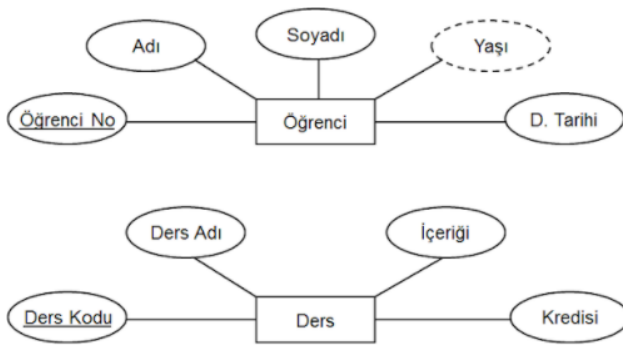
**VARLIK(ENTITY):** Var olan ve diğer nesnelerden ayırt edilebilen bir nesnedir.Örneğin, bir bilgisayar, bir kedi, bir kişi varlık olarak ifade edilir.

Varlıklar arasındaki bağlantı **ilişki** olarak ifade edilir.

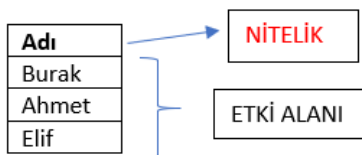
Bir bankanın müşterileri ve bu müşteriler için bankanın düzenlediği hesaplar birer varlıktır. Bu iki varlık arasında **müşteri - hesap** no biçiminde ifade edilen bir ilişki vardır.

**Öğrenci** bir varlıktır.Ve **Öğrenci** varlığının **Nitelikleri**, Öğrenci NO, Adı, Soyadı, Yaşı,D. Tarihi 'dir.

**Ders** bir varlıktır. Ve **Ders** varlığının **Nitelikleri**, Ders Kodu,Ders Adı,Ders İçeriği, Kredisi'dir.



Niteliğin aldığı değerlere **etki alanı (domain)** adı verilir.



Sütunlar (nitelikler)			Satırlar (Kayıtlar)
No	Adı	Bölüm No	
25	AHMET	10	
13	SENE	10	
28	ORKUN	30	

İlişkisel Veritabanı: *İlişkisel veritabanı*, birbiriyle ilişkili veri noktalarını depolayan ve bunlara erişim sağlayan bir veritabanı türüdür. İlişkisel veritabanı, ilişkisel modeli temel alır. Farklı tablolara yerleştirilmiş olan verilerin, belirli alanlara göre ilişkilendirildiği modellerdir. (<https://www.oracle.com/tr/database/what-is-a-relational-database/>)



Yukarıda **Futbolcular** tablosunda bulunan futbolcu\_id alanı ile, **Takim Futbolcu** tablosunda ilişki kurularak veri erişimi sağlanmıştır. Benzer şekilde **Takimler** tablosu ile, **Takim Futbolcu** tablosu üzerinden **Futbolcular** tablosu ile ilişki kullanılarak verilere erişim sağlanabilmektedir.

## İLİŞKİSEL CEBİR

İlişkisel Cebir, biçimsel olarak kullanılan bir sorgu yapısıdır. Burada yazılan cebir ifadeleri, daha sonra SQL komutlarına dönüştürülmektedir.

**SELECT(SEÇME) İŞLEMİ:**  $\sigma$  (sigma) işareti ile gösterilmektedir. Kayıtların seçilme işlemi için kullanılır. Seçim karşılaştırma ifadeleri  $=, >, <, \neq, \leq, \geq$  olarak kullanılmaktadır. Mantıksal operatörler olan “ve” için  $\wedge$ , “veya” için  $\vee$  ifadeleri kullanılmaktadır.

$\sigma_{\text{secim\_kriteri}}(\text{Tablo})$

## Öğrenci

Oğrenci_No	Ad	Soyad	Bolum
12	Ali	Ay	Bilgisayar Programcılığı
24	Cem	Yıldız	Fizik
27	Mutlu	Arı	İşletme
35	Burak	Bozkurt	Fizik
41	Uğur	Kaya	Bilgisayar Programcılığı

$\sigma_{\text{Bolum}=\text{"Fizik"}}(\text{Öğrenci}) \rightarrow$  Çıktı: Öğrenci tablosunda, Fizik Bölümündeki kayıtları seçme

$\sigma_{\text{Bolum}=\text{"Fizik"} \wedge \text{Ogrenci\_No} > \text{"24"}}(\text{Öğrenci}) \rightarrow$  Çıktı: Öğrenci tablosunda, bölümü fizik olan ve öğrenci numarası 24 'ten büyük kayıtları seçme.

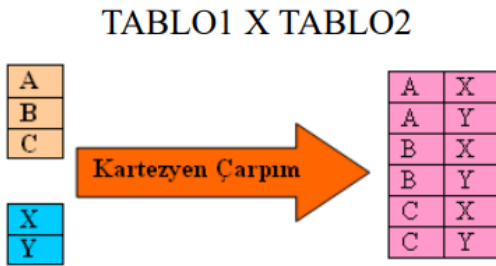
**PROJEKSİYON(PROJECT) / ATMA İŞLEMİ** : Bu işlem  $\Pi$  Sembolüyle gösterilir. Bir ilişkiyi belirli nitelikleri atmak için kullanılır.

$\Pi_{\text{Sutun\_isimleri}}(\text{TABLO})$

$\Pi_{\text{Ad,Bolum}}(\text{Öğrenci}) \rightarrow$  Çıktı: Öğrenci tablosunda Ad ve Bolum bilgilerini listeleme.

$\Pi_{\text{Ad,Bolum}}(\sigma_{\text{Bolum}=\text{"Fizik"}}(\text{Öğrenci})) \rightarrow$  Çıktı: Öğrenci tablosunda, Fizik bölümündeki öğrencilerin, Ad ve Soyad Bilgilerini getirir.

**KARTEZYEN(ÇARPMA) İŞLEMİ**: X sembolü ile gösterilir. İki ilişkideki bilgileri birleştirmek için kullanılır.



## Dersler

Ders	Kredi
Veritabanı YS	3
Nesne Yönelimli Programlama	4

Öğrenci ve Dersler tablosunun birleşimi ile bilgisayar programcılığı bölümüne olan öğrenciler ve aldığı derslerin listelemeesi,

$\sigma_{\text{Bolum}=\text{"Bilgisayar Programcılığı"}}(\text{Öğrenci X Dersler}) \rightarrow$  Çıktı:



Ogrenci_No	Ad	Soyad	Bolum	Ders	Kredi
12	Ali	Ay	Bilgisayar Programcılığı	Veritabanı YS	3
12	Ali	Ay	Bilgisayar Programcılığı	Nesne Yönelimli Programlama	4
41	Uğur	Kaya	Bilgisayar Programcılığı	Veritabanı YS	3
41	Uğur	Kaya	Bilgisayar Programcılığı	Nesne Yönelimli Programlama	4

**Birleştirme İşlemi(Union):**  $\cup$  Simgesi ile ifade edilir. İki ilişkiden birinde veya her ikisinde bulunan kayıtların seçilmesi için kullanılır.

KREDİ tablosu

Müşteri	Bakiye	İlçe
Burak	100	Beşiktaş
Begüm	200	Şişli
Sezin	150	Ulus

MEVDUAT tablosu

Müşteri	Bakiye	İlçe
Burak	50	Beşiktaş
Selin	125	Beşiktaş
Sezin	300	Ulus
Begüm	100	Ulus

Bankanın Beşiktaş şubesinde mevduat ve/veya kredi hesabı bulunan kişileri seçmek ve sadece isimlerini sunmak için :

$$\prod_{Kredi.Müşteri} (\sigma_{İlçe="Beşiktaş"} (KREDİ)) \cup \prod_{Mevduat.Müşteri} (\sigma_{İlçe="Beşiktaş"} (MEVDUAT))$$

Müşteri
Burak
Selin

**KESİŞİM İŞLEMİ(Set-INTERSECTION):** İlişkilerden ikisinde de bulunan kayıtları seçmek için kullanılır.  $\cap$  Sembolü ile ifade edilir.

Bankanın Beşiktaş şubesinde hem mevduat hem de kredi hesabı bulunan kişileri seçmek ve sadece isimlerini listele.

$$\prod_{Kredi.Müşteri} (\sigma_{İlçe="Beşiktaş"} (KREDİ)) \cap \prod_{Mevduat.Müşteri} (\sigma_{İlçe="Beşiktaş"} (MEVDUAT))$$

Müşteri
Burak

**Fark İşlemi(Set-Difference):** “-” Sembolü ile ifade edilir. İlişkilerden birine bulunup, diğerinde bulunmayan kayıtları seçmek için kullanılır.

Bankanın Ulus şubesinde mevduat hesabı olan, ve Ulus şubesinde kredi hesabı olmayan kayıtlar için,

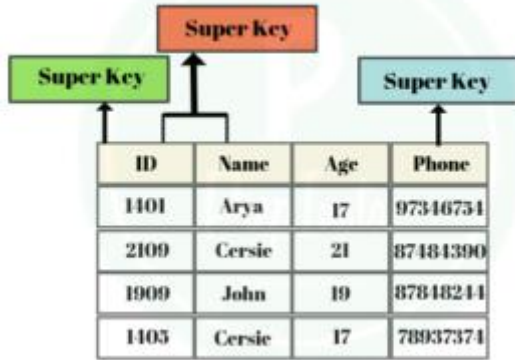
$$\prod_{Mevduat.Müşteri} (\sigma_{İlçe="Ulus"} (MEVDUAT)) - \prod_{Kredi.Müşteri} (\sigma_{İlçe="Ulus"} (KREDİ))$$

Müşteri
Begüm

## ANAHTARLAR(KEYS)

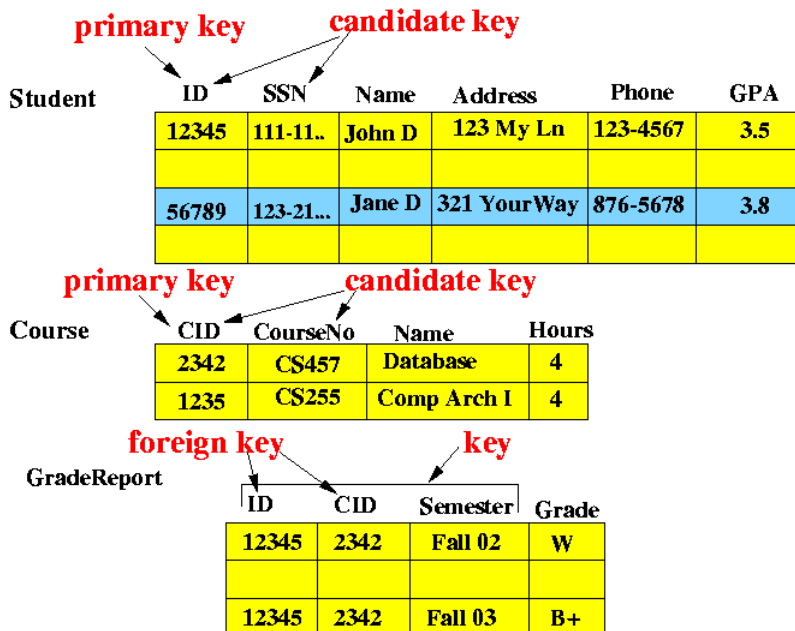
Varlıkları ayırt etmek için kullanılan niteliklere varlık kümesinin **anahtarı(key)** denir.

**Süper anahtar**: Bir varlığı kesin olarak tanımlamaya yarayan nitelik'e/anahtara süper anahtar denir. Bu anahtar tek bir nitelikten oluşacağı gibi birden fazla nitelikten de oluşabilir.



**Aday Anahtar(Candidate KEY)**: Varlık kümesi içinde var olan bir varlığı kesin olarak tanımlamaya yarayan anahtarlara denir.

**Birincil Anahtar(Primary Key)**: Aday anahtarlardan bir tanesinin seçilmesi ile birincil anahtar oluşturulur. Primary Key, Unique kısıtlaması, ve Not Null ile belirlenen alana, tablodaki her kayıt için birbirinden farklı olarak bir değer tanımlanmasını sağlar.



## KAYNAKLAR

[https://www.tutorialspoint.com/dbms/dbms\\_data\\_schemas.htm](https://www.tutorialspoint.com/dbms/dbms_data_schemas.htm)

Görsel Kaynağı İlişkisel model <http://www.zskblog.com/detay.aspx?id=21>

<http://www.ikslab.com/iliskisel-cebir>

[http://user.ceng.metu.edu.tr/~ocoban/research/bilgisayar\\_nedir.doc](http://user.ceng.metu.edu.tr/~ocoban/research/bilgisayar_nedir.doc)

<https://qastack.info.tr/programming/7284/what-is-turing-complete>

Görsel Kaynağı <https://prepinsta.com/dbms/super-key/>

<http://www.mathcs.emory.edu/~cheung/Courses/377/Syllabus/3-Relation/intro2.html>

Yrd. Doç. Dr. Fatih YÜCALAR -Celal Bayar Üniversitesi Yazılım Mühendisliği-Veritabanı Yönetim Sistemleri Ders Notları

[www.yazilimcildunyasi.com](http://www.yazilimcildunyasi.com)

[www.javayaz.com](http://www.javayaz.com)