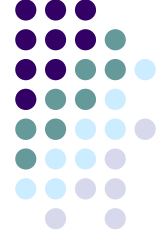


# VERİTABANI DERS NOTLARI

Yrd.Doç.Dr. Buket Dođan



1



## Ders İeriđi

- Veritabanı ve ilişkisel veritabanı kavramı, tasarımı ve yönetimini anlamak,
- veri tabanı sistemlerinin denetimi ve erişimi yöntemlerini ve araçlarını öğrenmek, (SQL komutlarının kullanımı)
- verilecek teori bilgiler temelinde VTYS uygulamalarını (Microsoft Access, SQL Server programları ile) yapmaktır.



2



## TEMEL KAVRAMLAR



- Veri
- **Olguların, kavramların, veya talimatların,** insan tarafından veya otomatik yolla iletiřim, yorumlama ve iřleme amacına uygun bir biimde ifadesidir.
- Genellikle, biz veri veya veri birimleri  zerindeki iřlemlerimizi **varlık hakkında her hangi bilgi almak iin** gerekleřtiririz.
- Veri kaydedilebilir bilinen gereklerdir.
-  rneėin bir kiřinin ismi, adresi, telefon numarası gibi.

3



## VERİTABANI NEDİR



- **Veri tabanı**
  - D zenli bilgiler topluluėudur.
  - Bilgisayar ortamında saklanan d zenli verilerdir.
  - **Bilgisayar terminolojisinde, sistematik eriřim imkanı olan, y netilebilir, g ncellenebilir , tařınabilir, birbirleri arasında tanımlı iliřkiler bulunabilen bilgiler k mesidir.**
  - Bilgisayarda sistematik Őekilde saklanmış, programlarca iřlenebilecek veri yıėınıdır

4



## Veri Tabanı Yönetim Sistemi- VTYS



Veri tabanı tanımlamak, yaratmak, yaşatmak ve veri tabanına denetimli erişim sağlamak için kullanılan yazılım sistemidir.

5



## TEMEL KAVRAMLAR



### KLASİK DOSYA YAPILARI

- Veri saklama birimlerinde depolanan veri topluluklarına “dosya” adı verilmektedir.
- Dosyalar ise kendi içersinden kayıtlara bölünmüştür.
- Örneğin öğrencilerin bilgilerinin tutulduğu bir dosyayı düşünelim:

6

## TEMEL KAVRAMLAR



ALAN 1	ALAN 2	ALAN 3
ADI	BABA ADI	DOĞUM YERİ
ADI	BABA ADI	DOĞUM YERİ
ADI	BABA ADI	DOĞUM YERİ
⋮		
} <b>TABLO</b>		

→ KAYIT

Kayıtlar birbiri ile ilişkili alanlardan(field) oluşmaktadır.

Her kayıt farklı bilgileri içermektedir.

7

## DOSYA SİSTEMLERİNİN SAKINCALARI



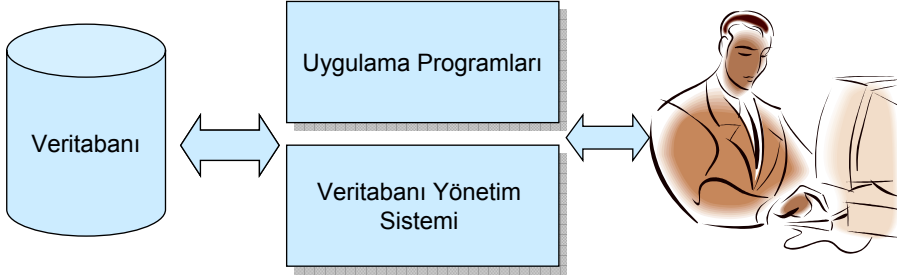
- Klasik dosya sistemleri kullanılmaya başlandıktan sonra bazı dezavantajları ortaya çıkmıştır. Bunlar şöyle sıralanabilir :
- **Veri tekrarı:** Aynı veri çeşitli dosyalarda birden fazla yer alabilmektedir buda sistemin hantallaşmasına neden olur. Mesela bir stok dosyasında stok numarası verisinin malzeme dosyasında, fatura dosyasında ve ambar girişi dosyasında yer alması gibi.
- **Verinin birkaç dosyada güncellemesi:** Veri birden fazla dosyada tekrar edilebildiği için, verinin bir dosyada güncellenip diğerlerinde güncellenmemesi Veri Bütünlüğünün (Data Integrity) bozulmasına neden olabilir. Buna bağlı olarak birbiri ile çelişen raporlar üretilebilir.
- **Belleğin tekrarlı bilgi nedeniyle israfı:** Aynı verinin birden fazla dosya içinde bulunması nedeniyle kullanılan veri hard diskte fazla yer işgal edecek. Yani hard disk tekrarlı veriler için kullanılmış olacaktır.
- **Sadece belirli bir dilin kullanılması :**Verilerin dosya sisteminde saklandığı ortamlar için değişik programlama dillerinden bir tanesi kullanılır. Kullanılan bu programlama dili ise SQL dili gibi esnek değildir.

8

# VERİTABANI SİSTEMLERİ



- Veritabanı sistemleri, veri kümelerinin düzenli biçimde tutulduğu ve bu verilerin yazılımlar aracılığı ile yönetildiği ortamlardır.



# VERİTABANI SİSTEMLERİ



- VYS'ler aşağıdaki bilgileri barındırmaktadır
  - İlişkili olan veriler (Collection of interrelated data)
  - Veriye ulaşmak için gerekli olan yazılımlar kümesi
- Veritabanı Uygulamaları (Database Applications)
  - Bankalar: tüm işlemler / hareketler
  - Havayolları: rezervasyonlar, zaman programları
  - Üniversiteler: Kayıt, notlar
  - Satış: müşteriler, ürünler, alımlar
  - Çevrimiçi Perakencileri: Sipariş Kayıtlar, Kişiselleştirilmiş tavsiyeler.
  - Üretim: imalat, stok, siparişler, tedarik ihtiyaçları
  - İnsan Kaynakları: personel kayıtları, maaşlar, vergi kesintileri
- Veritabanları hayatımızın her alanında kullanılmaktadır.



## Veritabanı Sistemlerinin Üstünlükleri

- Verinin tekrarlanması önler.
- Veritabanı sistemleri alt sistemler arasında ilişki kurulması ve birçok uygulamada verilerin aynı veritabanı içerisinde ortak olarak tasarlanmasını öngörür.
- Verilerin tutarlı olmasını sağlar.
- Veri bütünlüğü(data integrity), verinin doğruluğunu ve tutarlılığını ifade etmektedir. Veri girişlerine kısıtlar konularak sadece istenilen aralıkta değer girişi sağlanabilir.



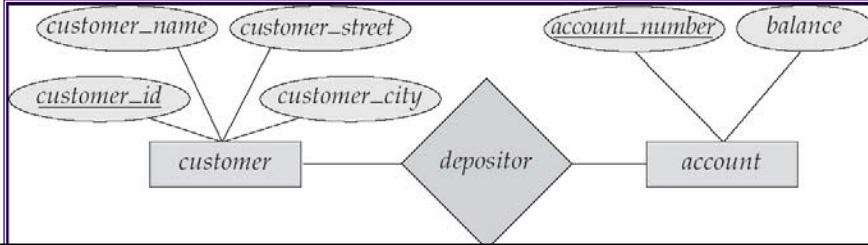
## Veritabanı Sistemlerinin Üstünlükleri

- Aynı andaki erişimlerde tutarsızlıkların ortaya çıkmasını önler.
- Veritabanı uygulamalarında, veritabanı nesnelere başka uygulamalar ve farklı kullanıcılar tarafından paylaşılabilir.
- Verilerin güvenliğini sağlar.
- Her kullanıcının erişeceği veriler ayrı ayrı tanımlanabilir. Yetkiler ve kısıtlamalar ile istenilen kullanıcı erişim ayarları gerçekleştirilir.

## Varlık-İlişki Modeli (The Entity-Relationship Model)



- Veri çözümleme ve modellemede ilişkilerin ortaya konması için kullanılan araçtır.
  - Varlık (Entity): Bir alan içerisinde diğer nesnelere ayırt edilebilen bir şey ("thing") yada "nesne" ("object")
- Niteliklerin kümesi (set of attributes) tarafından tanımlanır.
  - İlişki (Relationship): Birden fazla varlığın arasındaki bağıntı-ilişki.
- Görsel olarak varlık-ilişki tablosu ile gösterilir:



## Varlık-İlişki Modeli



- Varlık(Entity): Var olan ve diğer varlıklardan ayırt edilebilen nesnedir. (Bir kitap, öğrenci, veritabanı dersi birer varlıktır.)
- Varlık Dizisi: Aynı türdeki varlıklar varlık kümesini oluştururlar. Bir okuldaki tüm öğrenciler "öğrenci" isimli varlık kümesi olarak değerlendirilir.

## İlişki ve İlişki Kümeleri



- Varlıklar arasındaki bağlantıya ilişki adı verilir.örneğin “Burak” varlığı ile “Dersler” varlığı arasından ilişki vardır.
- İlişki kümesi, aynı türdeki ilişkilerin kümesidir, bu ilişki kümesi R ile gösterilir.
- $E_1, E_2, \dots, E_n$  varlık kümeleri, R ise ilişkiyi tanımlamaktadır.

## İlişki ve İlişki Kümeleri



- $E_1 = \{\text{Ayşe, Burak}\}$
- $E_2 = \{\text{Elektronik, İngilizce}\}$
- Bu iki küme arasındaki ilişki, öğrenciler ve dersler arasındaki ilişkidir. Tüm öğrencilerle tüm dersler arasındaki ilişki kartezyen çarpımı yapılarak ifade edilir.
- $E_1 \times E_2 = \{(\text{Ayşe, Elektronik}), (\text{Ayşe, İngilizce}), (\text{Burak, Elektronik}), (\text{Burak, İngilizce})\}$
- İki veri kümesi arasındaki geçerli tüm ilişkiler, R ilişki kümesinin bir alt kümesidir.



## İlişki ve İlişki Kümeleri



Müşteri no	Müşteri adı	Hesaplar	Bakiye
101	Ayşe	33344	1.000,00 YTL
203	Mehmet	33567	2.500,00 YTL
405	Derya	33790	45.000,00 YTL
607	Selin	34013	5.000,00 YTL

$R1 = \{(Ayşe, 33344), (Mehmet, 33567)\}$

$R2 = \{(Derya, 33790)\}$

$R3 = \{(Selin, 34013)\}$

## Nitelikler

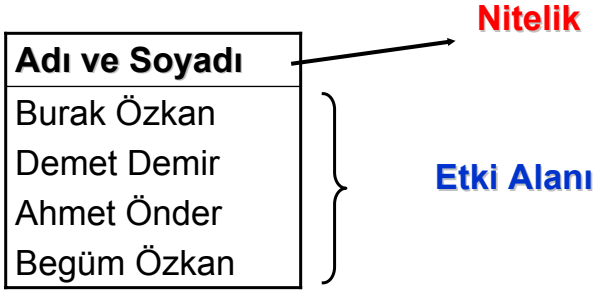


- Bir varlık çok sayıda nitelik yardımıyla tanımlanabilir. Örneğin, personel varlığının nitelikleri şu şekilde olabilir:
  - Personel No
  - Adı ve Soyadı
  - Adres
  - SSK no
  - Gelir



## Etki Alanı

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



## Türetilen Nitelik

- Bir nitelik kullanılarak, bir başka varlık nitelik elde edilebiliyorsa bu yeni niteliğe “türetilen nitelik” adı verilir.
- Örneğin personel varlığının “doğum tarihi” niteliğinden yararlanılarak “yaş” niteliği elde edilebilir.



## Çok Değere Sahip Nitelik

- Bir nitelik birden fazla değer ile eşleşebiliyor ise, “çok değere sahip nitelik” adı verilir.
- Örneğin, öğretmen varlığının **dersler** niteliği birden fazla değeri kapsar. Bir öğretmen birden fazla derse girmektedir.
- Öğrenci varlığının **okuduğu kitaplar** niteliği birden fazla kitabı kapsayabilir.



## Birleşik Nitelik

- Birden fazla nitelik birleştirilerek, yeni bir nitelik oluşturulabilir. Bu tür niteliklere birleşik nitelik denir
- Örneğin personelin “cadde” ve “şehir” nitelikleri birleştirilerek “ADRES” isimli yeni bir nitelik oluşturulabilir.

## Varlıklar arası İlişkiler(Eşleme)

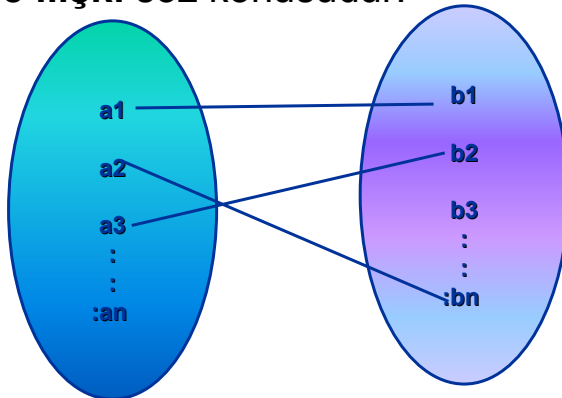


- Bir varlıkla ilişkiye girilebilecek varlıkların sayısına eşleme sayısı adı verilir.
- Eşleme sayısı  $n \geq 2$  varlık için söz konusudur ve ikili ilişkilerin ortaya konulması açısından yararlıdır.
- A ve B gibi iki varlık kümesi arasındaki R ilişki kümesi için eşleme durumları şu şekilde ifade edilir:
  - Birden-bire (One to One)
  - Birden-çoğa(One to Many)
  - Çoktan-bire (Many to One)
  - Çoktan-çoğa (Many to Many)

## Birden-bire İlişki



- A varlık kümesi içindeki bir varlık, B kümesi içindeki **sadece** bir varlık ile ilişkili ise **birden-bire ilişki** söz konusudur.



## Birden-bire ilişki



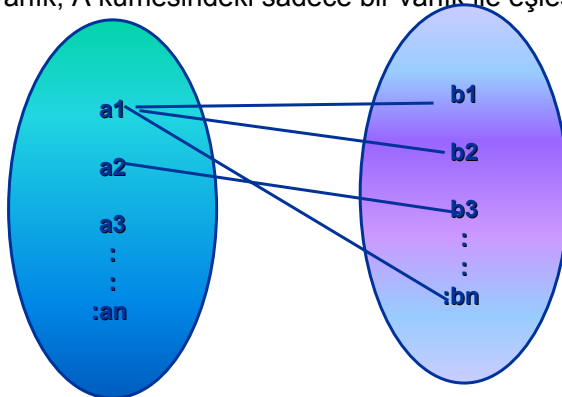
Müşteri no	Müşteri adı	Hesaplar	Bakiye
101	Ayşe	33344	1.000,00 YTL
203	Mehmet	33567	2.500,00 YTL
405	Derya	33790	45.000,00 YTL
607	Selin	34013	5.000,00 YTL

Her müşterinin bir hesabı olabilir.

## Birden-çoğa ilişki



- A kümesi içindeki bir varlık B kümesi içindeki birden fazla varlık ile ilişkili ise, bu ilişkiye birden-çoğa ilişki adı verilir. B kümesindeki bir varlık, A kümesindeki sadece bir varlık ile eşleşebilir.



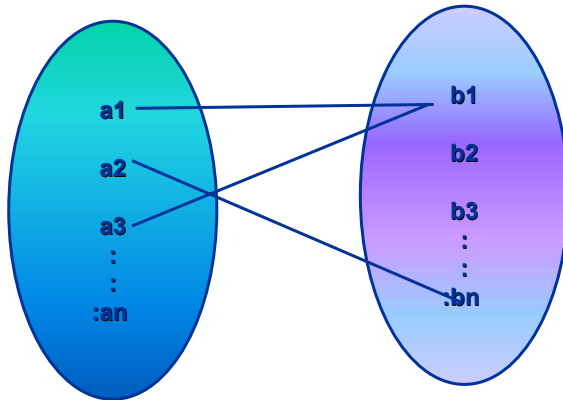
## Birden-çoğa İlişki

ID	Öğretmen adı	NO	OGR_ID	Girdiği DERS
101	Ayşe	1	101	Matematik
203	Ahmet	2	101	Geometri
405	Derya	3	405	Bilgisayar Prog.
607	Selin	4	405	Office Programları

Öğretmenler birden fazla derse girmektedir.

## Çoktan-bire İlişki

- A varlık kümesindeki birden fazla varlık, B kümesindeki bir varlık ile ilişkili ise bu eşleşmeye çoktan-bire ilişki adı verilir.



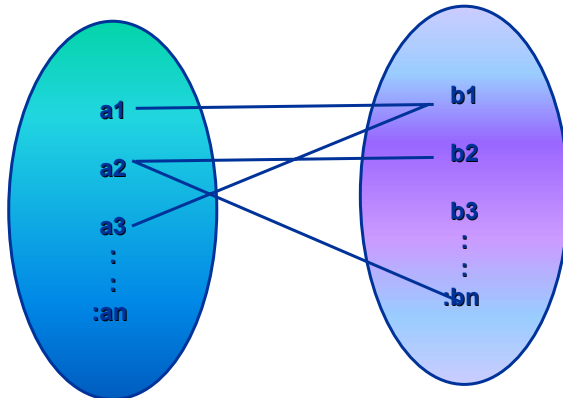
## Çoktan-bire İlişki

No	Kişi_ID	DERS
1	101	Matematik
2	101	Geometri
3	405	Bilg. Prog.
4	405	Office Prog

ID	İsim	Doğum.Tarihi
101	Ayşe	3.06.1990
203	Ahmet	12.04.1980
405	Derya	15.04.1983
607	Selin	5.07.1981

## Çoktan-Çoğa İlişki

- A varlık kümesindeki birden fazla varlık, B kümesindeki birden fazla varlık ile ilişkili ise bu eşleşmeye çoktan-çoğa ilişki adı verilir.





## Çoktan-Çoğa İlişki

- Çoktan-çoğa ilişki en genel ilişki biçimidir. Bu ilişki herhangi bir sınırlamanın olduğu durumlar için geçerli olacaktır.
- Müşteri-hesap ilişkilerinde aile üyelerinin ortak hesap açabilmesi durumunda çoktan-çoğa ilişki söz konusu olacaktır.

## Varoluş Koşulu



- Eğer bir X varlığının bulunması Y varlığının bulunmasına bağlı ise, X'in Y'ye bağlı olduğundan söz edilebilir.
- Y silinirse, X'in bir anlamı kalmayacaktır.
- Bu durumda Y baskın varlık(dominant entity)
- X ise bağımlı varlık(subordinate entity) adı verilir.
  - Örneğin, bir müşterinin hesabı silineceğinde, bu müşterinin hesap hareketlerinin de silinmesi gerekmektedir. Hesap hareketleri, hesap varolmadan var olamaz.



## Anahtar



- Varlık kümesi içinde, varlıkları birbirinden ayırt etmek için kullanılan bu tür niteliklere varlık kümesinin anahtarı adı verilir. İki tür anahtar vardır.
- **Süper anahtar (superkey):**Varlık kümesinde yer alan bir varlığı kesin olarak tanımlamaya yarayan anahtara süper anahtar adı verilmektedir. Bu anahtar sadece bir nitelikten oluşabileceği gibi, birden fazla niteliğin birleşiminden de oluşabilir. Süper anahtarlar süper küme oluşturur. Bir süper anahtarın herhangi bir süper kümesi daima bir süper anahtar olarak kabul edilir.
- Örneğin SSK no süper anahtardır. Fakat isim alanı süper anahtar olamaz. SSK no ve isim alanı birlikte süper anahtar olarak kabul edilebilir.

## Anahtar



- **Aday anahtar (candidate key) :** Varlık kümesinde bir varlığı tanımlamaya yarayan bir başka anahtar türü aday anahtar dır.
- Bir varlık kümesinin süper anahtarı bir veya daha fazla niteliğin birleşiminden oluşabilmektedir.
- Aday anahtar ise, süper anahtar özelliklerine sahip tek nitelikli anahtardır.



## Anahtar

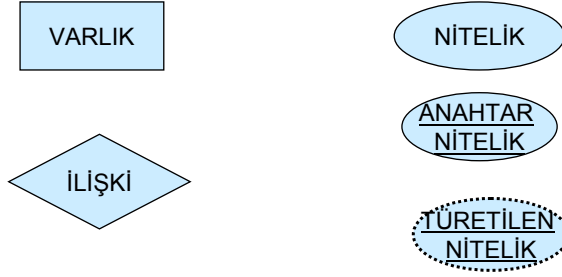
- Eger bir üniversitede tüm öğrencilerin numaraları birbirinden farklı ise öğrencileri belirlemek için öğrenci numarası yeterlidir.
- Bu durumda öğrenci numarası öğrenci varlık kümesi için aday anahtardır.
- İçinde öğrenci numarası bulunan her nitelik grubu ise(öğrenci numarası, adı, soyadı gibi) ise bu varlık kümesinin süper anahtardır.



## Varlık-İlişki Şemaları (Entity-Relationship Model)

- Varlık-ilişki modeli ; Veritabanı modelleri içerisinde , varlık ve bu varlıkların birbirleri arasındaki ilişkilere dayanarak herhangi bir ön-veri olmaksızın model oluşturmakta kullanılan modeldir.
- Buradaki varlık; benzersiz bir şekilde tanımlanabilen ve bağımsız var olabilme yetisine sahip nesne ya da oluşum olarak tanımlanabilir.
- Varlıklar , ev, araba gibi fiziksel nesnelere olabileceği gibi müşteri ödemesi veya sipariş gibi soyut nesnelere de içerirler

## Varlık-İlişki Şemaları

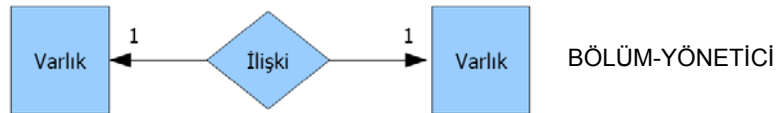


## Varlık-İlişki Şemaları

Varlıkların aralarında kurulabilecek ilişki türleri aşağıdaki gibi tanımlanır ve model olarak ifade edilir

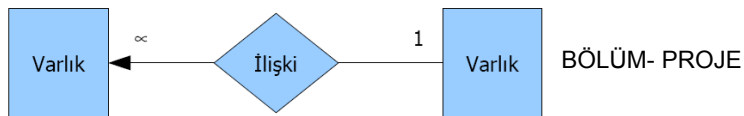


a- Birden-bire



Çizim 1: Birden-bire

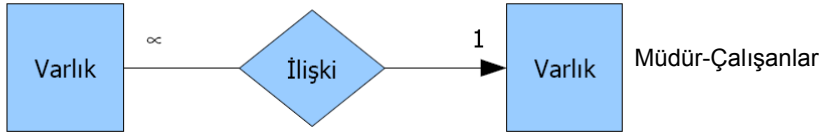
b- Birden-çoğa



Çizim 2: Birden-Çoğa

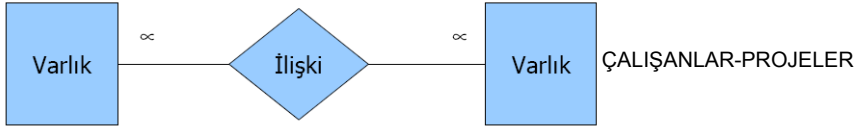
## Varlık-İlişki Şemaları

c- Çoktan-bire



Çizim 4: Birden-bire

d- Çoktan- çoğa

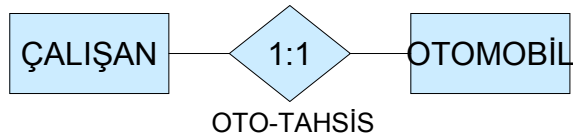


Çizim 5: Çoktan-çoğa

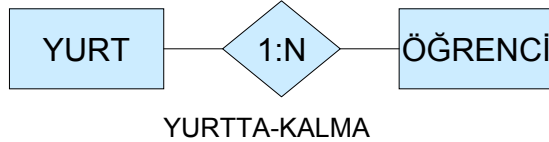
## Özelli İlişkiler

•İlişkinin *büyüklüğü* ile ilgilidir

Bire-bir:



Bire-çoklu:



Çoka-çoklu:

