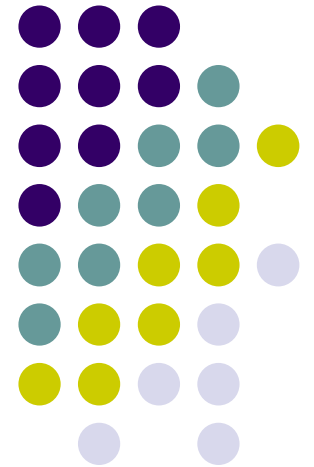


# Veri Tabanı Yönetimi

## 02 – Veri Modelleri

Altuğ Tanaltay

Not: Bu dersin Türkçe slaytları Trakya Üniversite'sinden Altan Mesut tarafından sağlanmıştır.



# Veri Modeli



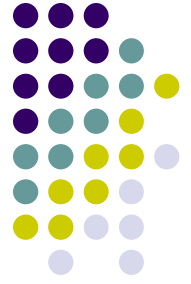
- Verileri mantıksal düzeyde düzenlemek için kullanılan yapılar, kavramlar ve işlemler topluluğuna veri modeli (data model) denir.
- Her VTYS belirli bir veri modelini kullanır.
- Bir VTYS'yi kullanarak oluşturulacak her veri tabanında yer alacak veriler ve veriler arası ilişkiler, mantıksal düzeyde ilgili veri modeline göre düzenlenir; bu veri modeli kullanılarak veri tabanının kavramsal ve dış şemaları oluşturulur.

# Veri Modelleri



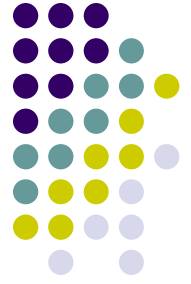
- Bugüne kadar geliştirilmiş olan çok sayıda veri modeli vardır.
- Ancak geçmişte ve günümüzde yaygın kullanılan veri modellerini 4 grupta toplamak mümkündür:
  - Sıradüzensel Veri Modeli (Hierarchical Data Model)
  - Ağ Veri Modeli (Network Data Model)
  - İlişkisel Veri Modeli (Relational Data Model)
  - Nesneye-Yönelik Veri Modeli (Object-oriented Data Model)
- Yukarıdaki sıralama aynı zamanda kronolojik bir sıralamadır.

# Veri Modelleri : Tarihçe



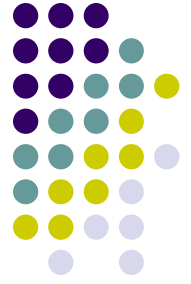
- Sıradüzensel (Hiyerarşik) Veri Modeli en eski model olup 60 ve 70'li yıllarda çok kullanılmıştır.
- 1969'da ortaya çıkan Ağ Veri Modeli 1970'li yıllarda ve 1980'li yılların ilk yarısında kullanılmıştır.
- İlişkisel veri modeli de ilk kez 1969 yılında ortaya atılmış, 1970'li yılların sonunda kullanılmaya başlanmış ve 1985 yılından sonra yaygınlaşmış bir yaklaşımdır.
- 1990'lı yıllarda yaygın kullanılan VTYS'lerin hemen hemen tümünün ilişkisel tabanlı olduğu söylenebilir.

# Veri Modelleri : Tarihçe

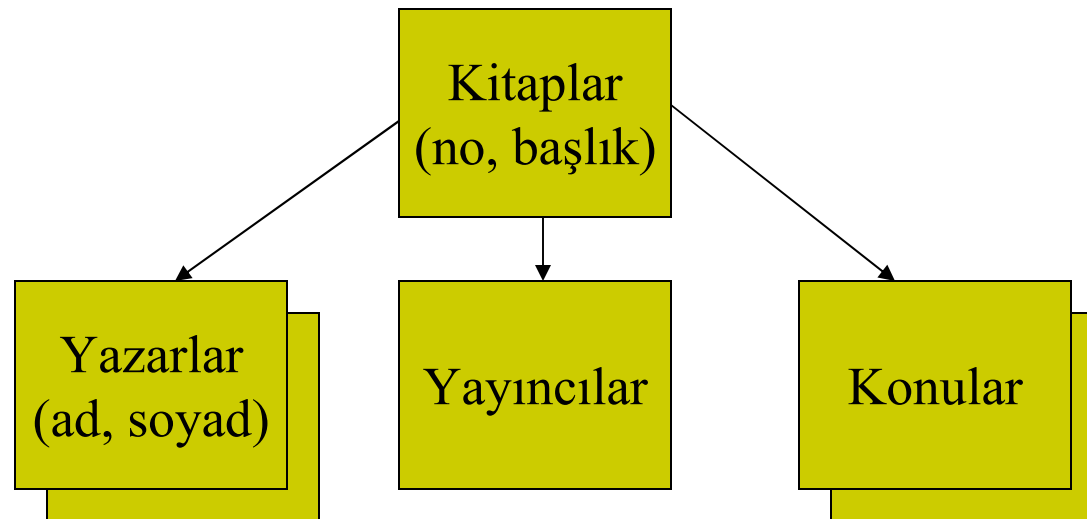


- Nesneye-yönelik veri modeli yaklaşımı ise on yılı aşkın süredir gündemde olan, günümüzde çok yaygın kullanılsa bile, kullanımı giderek yaygınlaşan bir yaklaşımdır.
- Geçmişe baktığımızda, ilişkisel yaklaşımın kullanılmaya başlanması ile sıradüzensel ve ağ yaklaşımlarının terk edildiği görülmektedir.
- Buna karşılık nesneye-yönelik yaklaşımın kullanılmaya başlanması ile ilişkisel yaklaşım terk edilmemiştir.
- Günümüzde hem ilişkisel hem de nesneye-yönelik yaklaşımı birlikte kullanan VTYS'lerinin yaygınlaştığı görülmektedir (ORDBMS).

# Sıradüzensel (Hiyerarşik) Model



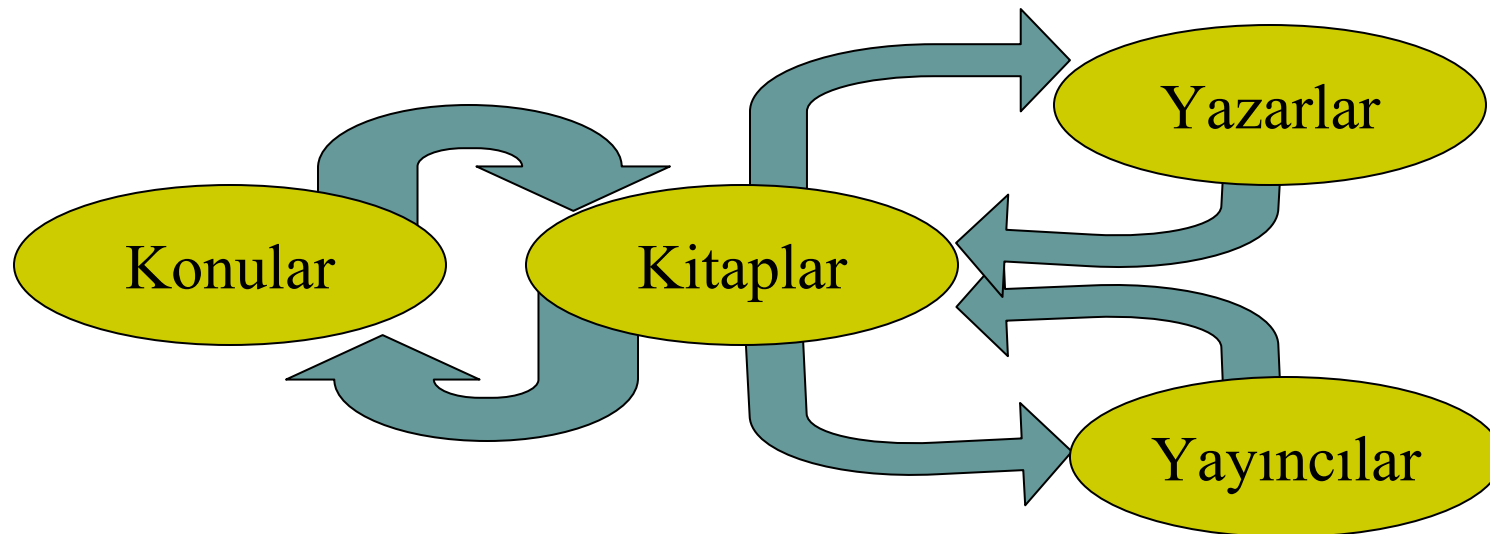
- 1960'lar ve 1970'ler
  - Ağaç veri yapısına benzer. Her kaydın bir ebeveyn kaydı, birçok çocuk kaydı var (IBM IMS: *Information Management System*).



# Ağ Modeli



- 1970'ler
  - Her kaydın birçok ebeveyn ve birçok çocuk kaydı bulunabilir. (TurboIMAGE, IDMS, RDM Embedded, RDM Server )



# İlişkisel Model



- 1980'ler
  - Veriler için kavramsal olarak basit bir model; veriler ve ilişkiler “tablolar” üzerinde tanımlanır ve tüm bilgiler görülebilecek şekildedir.

Book ID	Title	pubid	Author id
1	Introduction	2	1
2	The history	4	2
3	New stuff a	3	3
4	Another tit	2	4
5	And yet m	1	5

pubid	pubname
1	Harper
2	Addison
3	Oxford
4	Que

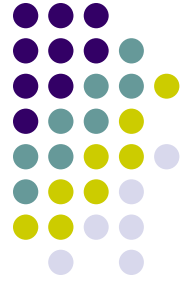
Authorid	Author nan
1	Smith
2	Wynar
3	Jones
4	Duncan
5	Applegate

Book ID	Subid
1	2
2	1
3	3
4	2
4	3

Subid	Subject
1	cataloging
2	history
3	stuff



# Nesneye-Yönelik Model



- 1990'lar
  - Veriler **nesne** olarak modellenir ve yaratılır.
  - Nesneye-Yönelik Programlama (OOP) da olan sınıf ve miras kavramlarına sahiptir.
  - Karmaşık veriler üzerinde işlem yaparken yüksek performans sunan bir yaklaşımdır.

# Nesneye-Yönelik İlişkisel Model



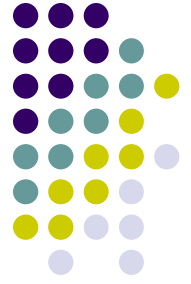
- 1990'lar
  - İlişkisel Modelin iyi bilinen özelliklerini Nesne Tabanlı özelliklerle birleştirir. Bunlar:
    - Kullanıcı tanımlı veri türleri
    - Kullanıcı tanımlı fonksiyonlar
    - Kalıtım ve alt sınıflar

# Varlık-İlişki Modeli (E-R Modeli)



- Varlık-ilişki modeli, ya da kısaca E-R modeli (Entity-Relationship model) 1976 yılında P.P. Chen tarafından geliştirilen bir modeldir.
- Bugüne kadar varlık-ilişki modeline dayalı hiçbir VTYS geliştirilmemiştir.
- Buna karşılık varlık-ilişki modeli, VTYS'den bağımsız veri çözümlemede ve semantik veri modellemede en çok kullanılan modeldir.

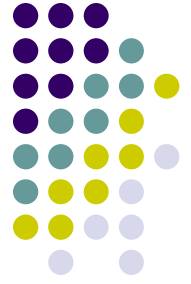
# Varlık-İlişki Modeli



- Bu model kullanılarak önce;
  - VTYS'den bağımsız olarak veriler çözümlenir,
  - veri modellemesi yapılır,
  - veriler ve veriler arası ilişkilerin anlamları ve özellikleri incelenerek E-R çizelgeleri oluşturulur;
  - kullanılacak VTYS belirlenir

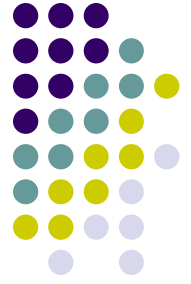
sonra da E-R çizelgeleri bu sistemin veri modeline dönüştürülerek veri tabanı şemaları oluşturulur.

# Varlık ve Varlık Kümesi



- Var olan ve benzerlerinden ayırt edilebilen her nesneye **varlık** (entity) denir.
  - Bir öğrenci, veri tabanı dersi, belirli bir kitap, Burak birer varlık olarak değerlendirilir.
- Aynı türden benzer varlıkların oluşturduğu kümeye ise **varlık kümesi** (entity set) adı verilir. Varlık kümeleri iç içe, kesişen ya da ayrık kümeler olabilir.
  - Öğrenciler, kız öğrenciler, Bilgisayar Mühendisliği Bölümü öğrencileri, yurttan kalan öğrenciler, renkler, dersler, yıllar, tarihler, satış miktarları,... vb. varlık kümesi örnekleri olarak sayılabilir.

# Nitelik



- Bir varlık kümesindeki varlıkların özelliklerini göstermek ve varlıkları birbirinden ayırt etmek için nitelikler (attributes) kullanılır.
- Gerçek dünyada varlık kümelerinin çok sayıda nitelikleri olabilir, ancak veri modellemede, gerçek dünyanın soyut bir modeli oluşturulduğu için, bu niteliklerin yalnız küçük bir kısmı, uygulamalar için gerekli olanları (örn. Sicil no, Ad, Soyad, Adres, ... vb) seçilerek kullanılır.

# Etki Alanı (Değer Alanı)



- Her niteliğin bir etki alanı (domain) vardır. Etki alanı ilgili niteliğin olabilecek değerlerinin tümünü içeren bir kümedir.

Adı ve Soyadı	→ Nitelik
Burak Özkan	} Etki Alanı
Begüm Özkan	
Dilay Dondurmacı	

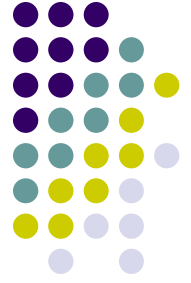
# Türetilen Nitelik



- Bir nitelik kullanılarak bir başka varlık niteliği elde edilebiliyorsa, bu yeni niteliğe türetilen nitelik adı verilir.
- Örneğin, “personel” varlığının “doğum tarihi” niteliğinden yararlanılarak “yaş” niteliği elde edilebilir. Bu örnekte “yaş” niteliği türetilen nitelikler, tasarımda ayrıca tanımlanmasına gerek yoktur.

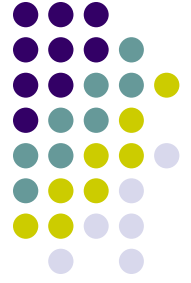


# Birleşik Nitelik



- Birden fazla nitelik birleştirilerek yeni bir nitelik oluşturulabilir. Bu tür niteliklere birleşik nitelik adı verilir.
- Örneğin, “mahalle”, “cadde”, “sokak”, “apartman”, “posta kodu” ve “şehir” gibi nitelikler birleştirilerek “adres” isimli yeni bir nitelik oluşturulabilir.

# İlişki



- Varlıklar arasındaki bağıntıya ilişki adı verilir.
- İkili ilişki:
  - bir öğrenci ile bir ders
  - bir firma ile bir malzeme
- Üçlü ilişki:
  - Bir işçi, bir ürün ve bir makine (işçi bu ürünü üretirken bu makineyi kullandığı için)

# İlişki Kümesi



- Aynı türdeki ilişkilerin oluşturduğu kümeye ilişki kümesi denir.
- Matematiksel olarak  $E_1, E_2, E_3, \dots, E_n$  varlık kümeleri arasındaki bir  $R$  ilişkisi aşağıdaki gibi tanımlanır:

$$R = \{(e_1, e_2, e_3, \dots, e_n) :$$

$$e_1 \in E_1, e_2 \in E_2, e_3 \in E_3, \dots, e_n \in E_n \}$$

# İlişki Kümesi : Örnek



- Aşağıdaki iki varlık kümesini göz önüne alalım:

$$E_1 = \{ \text{Ali, Ayşe} \}$$

$$E_2 = \{ \text{Matematik, Fizik} \}$$

- Bu varlık kümeleri için öğrenci ve aldığı ders ilişkileri aşağıdaki gibi ifade edilebilir:

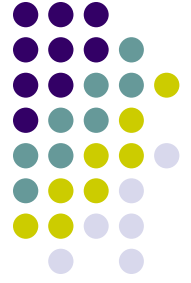
$$R_1 = \{ (\text{Ali, Matematik}) \}$$

$$R_2 = \{ (\text{Ali, Fizik}) \}$$

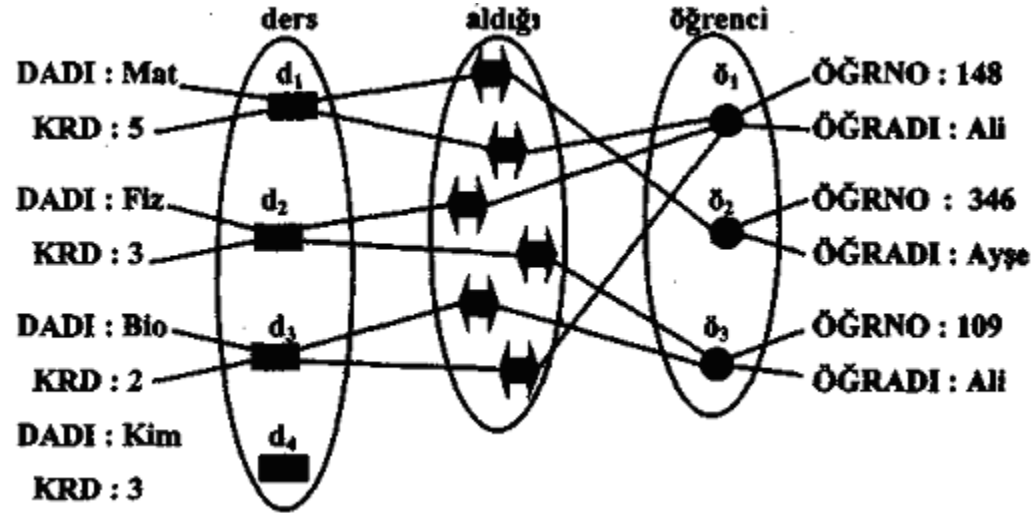
$$R_3 = \{ (\text{Ayşe, Matematik}) \}$$

$$R_4 = \{ (\text{Ayşe, Fizik}) \}$$

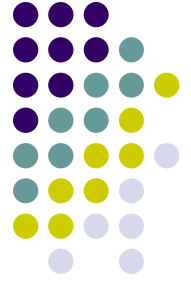
# İlişki Kümesi : Örnek



- Aşağıda şekilde “öğrenci” ve “ders” varlık kümeleri ile bu iki varlık kümesi arasındaki “aldığı” ilişki kümesi görülmektedir.

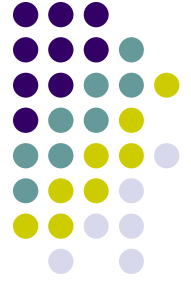


# Çoklu İlişki Kümesi

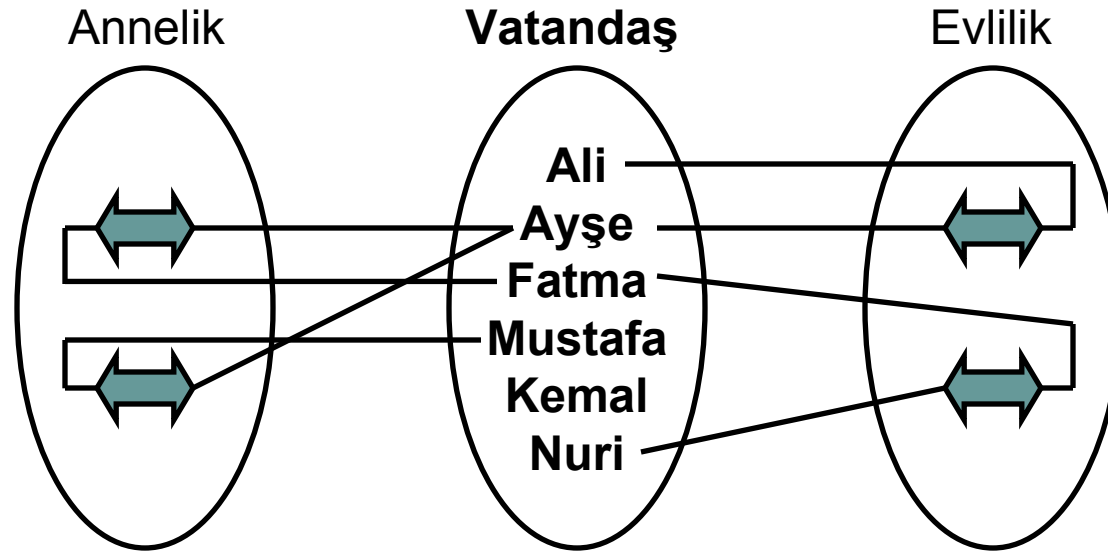


- İlişki kümeleri ikili, üçlü, dördlü, .. vb. olabilir.
- Ancak genellikle ikili ilişki kümeleri tercih edilir.
- Üçlü, dördlü, ..vb. ilişki kümesi yerine, istenirse birkaç ikili ilişki kümesi kullanılabilir. Örneğin;
  - Öğrenci, ders ve öğretmen varlık kümeleri arasındaki üçlü ilişki yerine 2 ya da 3 ikili ilişki kullanılabilir.
  - “Ali, Mehmet Hoca’nın anlattığı Fizik dersini alıyor” yerine “Ali Fizik dersini alıyor” ve “Mehmet Hoca Fizik dersini anlatıyor” kullanılabilir.

# Aynı Varlık Kümesi ile İlişki



- Aralarında ilişki tanımlanan varlık kümelerinden ikisi ya da daha çoğu aynı varlık kümesi de olabilir.



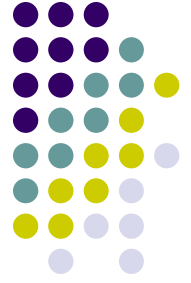
# Rol



- Aralarında ilişki kurulan varlıklardan her birinin ilişkideki işlevine varlığın rolü denir.
- Farklı varlık kümeleri arasındaki ilişkilerde roller dolaylı yoldan anlaşılabilirdiği için çoğunlukla açıkça belirtilmez
  - Örneğin, öğrenci ve ders arasında kurulan “aldığı” ilişkisinde varlıkların rolleri bellidir: öğrenci dersi alan, ders ise öğrenci tarafından alınandır.



# Rol : Örnek



- İlişkilerdeki roller belirlenirken, başka niteliklere de bakmak gerekebilir.
  - Örneğin “Personel” varlığında “ast-üst” ilişkisini belirlemek için “Görevi” niteliği dışında “Bölümü” niteliğine de bakmak gerekebilir.

## Personel varlığı:

Adı	Bölümü	Görevi
Burak	Muhasebe	İşçi
Begüm	Muhasebe	Yönetici
Dilay	Muhasebe	İşçi
Selin	Satış	Yönetici
Sezin	Satış	İşçi

## Yönetici (üst, ast) ilişkileri:

(Begüm, Burak)

(Begüm, Dilay)

(Selin, Sezin)

# İlişki Kümelerinin Sınırlandırılması



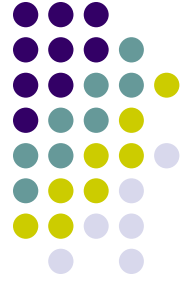
- İlişki kümeleri ile ilgili olarak bir dizi sınırlama tanımlanabilir.
- Bu sınırlamaların en önemlileri, aralarında ilişki kurulan varlık kümeleri arasındaki eşlemelerle ilgili sayısal sınırlamalardır.
- Bu tür sayısal sınırlamalar özellikle ikili ilişki kümeleri için çok önemlidir.

# İlişki Türleri

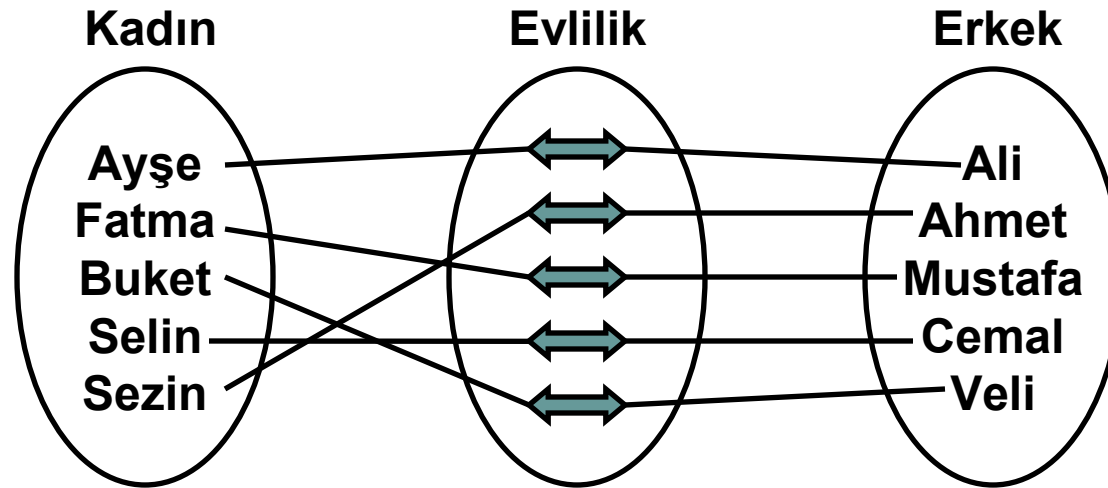


- A ve B varlık kümeleri arasında tanımlanan (A ve B varlık kümeleri aynı da olabilir), A'dan B'ye bir ilişki kümesi, eşleme sınırlamaları açısından aşağıdaki dört türden birinde olabilir.
  - Birden-bire (one-to-one)
  - Birden-çoğa (one-to-many)
  - Çoktan-bire (many-to-one)
  - Çoktan-çoğa (many-to-many)

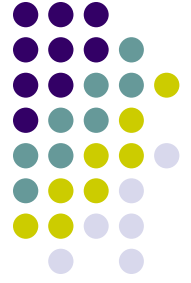
# İlişki Türleri : Birden-bire (1-1)



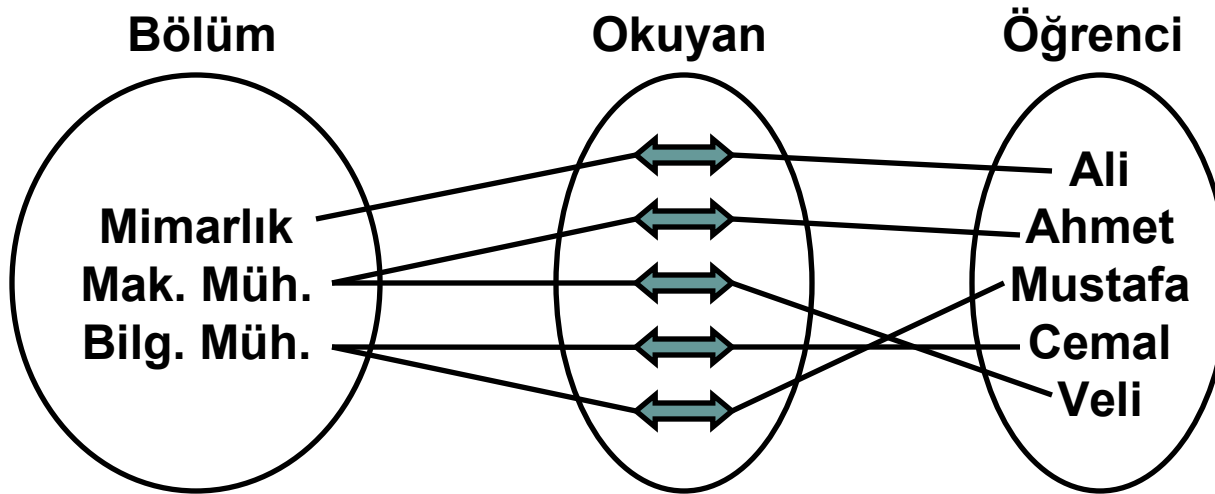
- Her a ile en çok bir b ve her b ile de en çok bir a arasında ilişki kurulabilir ( $a \in A, b \in B$ ).
- Örnek: “Evlilik” ilişkisi T.C. Medeni Kanunu’na göre birden-bire’dir.



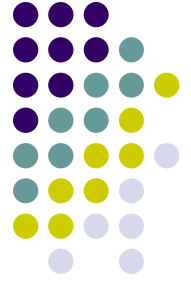
# İlişki Türleri : Birden-çoğa (1-n)



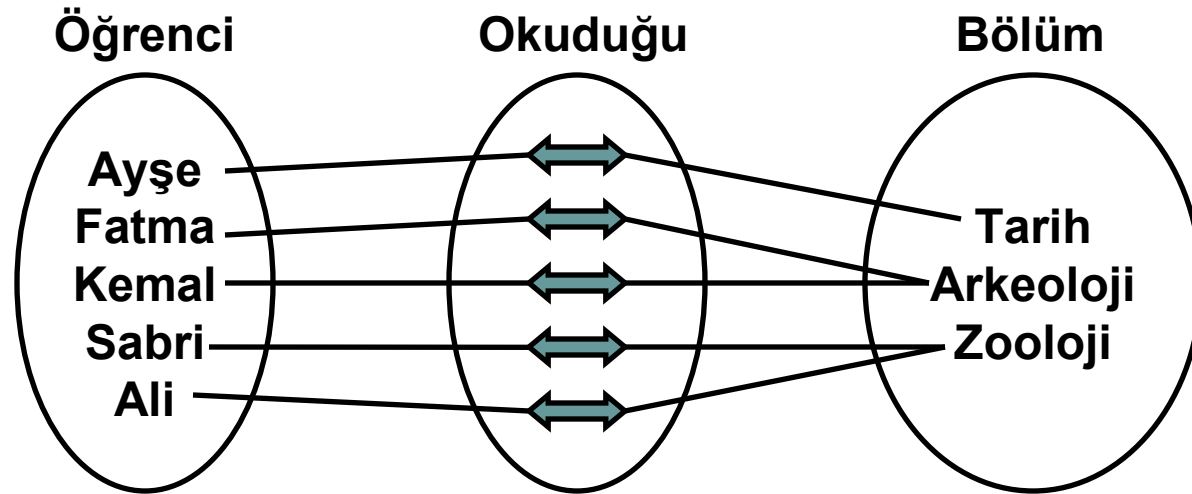
- Her a ile sıfır, bir veya birçok b ve her b ile de en çok bir a arasında ilişki kurulabilir.
- Örnek: “Bölüm” ve “Öğrenci” varlık kümeleri arasındaki “Okuyan” ilişkisi, bölümden öğrenciye doğru birden-çoğa şeklindedir.



# İlişki Türleri : Çoktan-bire (n-1)



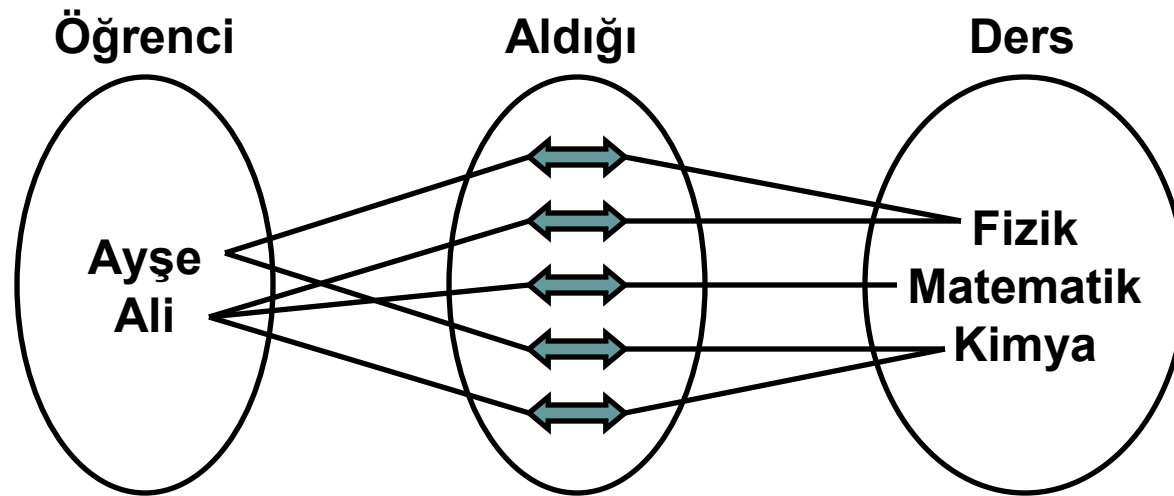
- Her a ile en çok bir b ve her b ile de sıfır, bir veya birçok a arasında ilişki kurulabilir.
- Örnek: “Öğrenci” ve “Bölüm” varlık kümeleri arasındaki “Okuduğu” ilişkisi, öğrenciden bölüme doğru çoktan-bire şeklindedir.



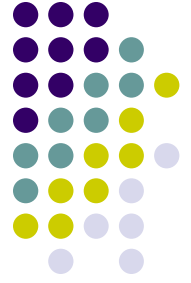
# İlişki Türleri : Çoktan-çoğa (n-m)



- Her a ile sıfır, bir veya birçok b ve her b ile de sıfır, bir veya birçok a arasında ilişki kurulabilir.
- Örnek: “Öğrenci” ve “Ders” varlık kümeleri arasındaki “Aldığı” ilişkisi, çoktan-çoğa şeklinde bir ilişkidir.



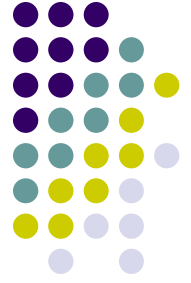
# İlişki Türünü Belirlerken...



- Yapılan modellemenin gerçek dünyaya uygunluğunu sağlaması için tanımlanan ilişki kümesinin türünün doğru belirlenmesi önemlidir.
- İlişki kümelerinin türü evrensel ve değişmez değildir. Gerçek dünya kurallarının bir yansıması olarak bir ilişkinin türü bir kurumdan diğerine değişebilir.
- İlişkinin türünün aynı kurum içinde zamanla değişmesi de olasıdır.



# Tür Seçimi için Örnek



- “Depo” ve “Malzeme” varlık kümeleri arasındaki “Bulunan” ilişkisi genelde çoktan-çoğa bir ilişkidir. Ancak eğer malzemeler sınıflandırılmış ve her sınıftaki malzemeler yalnız bir depoda bulunuyorsa bu ilişki depodan malzemeye bire-birçok şeklinde olacaktır.

# Varolma Bağımlılığı



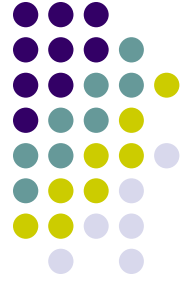
- “b” varlığının bulunması “a” varlığının bulunmasına bağlı ise, yani:
    - A ve B varlık kümeleri arasında birden-bire, ya da A'dan B'ye birden-çoğa bir R ilişkisi varsa ... VE ...
    - bir b'nin varolması bu b ile bir a arasında r ilişkisinin kurulmuş olmasına bağlı ise (r ilişkisi yüzünden bir a'ya bağlı olmayan b'ler var olamıyorsa)
- b a'ya varolma bağımlıdır denir.

# Varolma Bağımlılığı



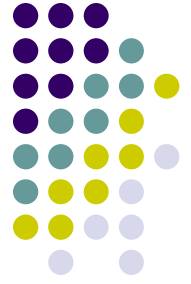
- Bu durumda;
  - a baskın (dominant) ya da birincil varlık,
  - b bağımlı (subordinate) ya da ikincil varlık olarak nitelenir.
  - ve a'nın silinmesi durumunda b'nin bir anlamı kalmayacaktır.

# Varolma Bağımlılığı için Örnek



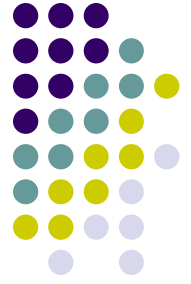
- Eğer ilgili öğretim kurumunda “bölümü belli olmayan öğrenci bulunamaz” kuralı geçerli ise, yani “her öğrenci mutlaka bir bölümün öğrencisi olmak zorunda” ise, öğrenci ve bölüm varlık kümeleri arasında varolma bağımlılığı vardır.
- Bu durumda bölüm üstün (dominant) varlık, öğrenci ise bağımlı (subordinate) varlıktır. İlgili öğretim kurumunda bir öğrencinin varolması ancak bir bölümün öğrencisi olması ile mümkündür.

# Anahtarlar



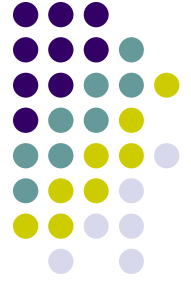
- Bir varlık kümesi içindeki varlıkları ya da bir ilişki kümesi içindeki ilişkileri birbirinden ayırt etmek için kullanılan nitelik ya da nitelik grubuna bu varlık ya da ilişki kümesinin anahtarı denir.
- Anahtar, hem varlık kümeleri hem de ilişki kümeleri için geçerli bir kavram olsa da, daha çok varlık kümeleri için kullanılır.

# Anahtar Türleri



- Süper anahtar (superkey)
  - Değerleri ile bir kümedeki varlıkları (veya ilişkileri) ayırt etmeyi sağlayan niteliğe (veya nitelik grubuna) bu varlık / ilişki kümesinin süper anahtarı denir. Ayırt etme özelliğine sahip olmak için gereğinden fazla nitelik içerebilir.
- Aday anahtar (candidate key)
  - Eğer bir varlık / ilişki kümesinin süper anahtarının bir altkümesi de bu varlık / ilişki kümesini ayırt edebiliyorsa, bu altküme aday anahtardır (ya da kısaca anahtardır).

# Anahtar Türleri için Örnek



- Eğer bir üniversitede tüm öğrencilerin numaraları birbirinden farklı ise, öğrencileri ayırt etmek için öğrenci numarası yeterlidir.
- Bu durumda “öğrenci numarası”, “öğrenci” varlık kümesi için aday anahtar ya da kısaca anahtardır, içinde öğrenci numarası bulunan her nitelik grubu (örneğin “öğrenci numarası”, “adı” ve “soyadı”) ise bu varlık kümesinin süper anahtarıdır.

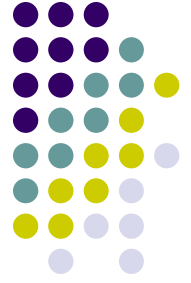
# Güçlü & Zayıf Varlık Kümeleri



- Her varlık kümesi için bir anahtar bulmak mümkün olmayabilir.
- Eğer bir varlık kümesinin niteliklerinden en az bir anahtar oluşturulabiliyorsa, bu varlık kümesine **güçlü** (strong) varlık kümesi denir.
- Eğer bir varlık kümesinin niteliklerinin tümü alınsa bile bir anahtar oluşturmuyorsa bu varlık kümesine **zayıf** (weak) varlık kümesi denir.

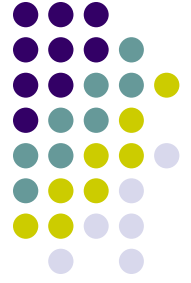


# Zayıf Varlığı Güçlendirmek



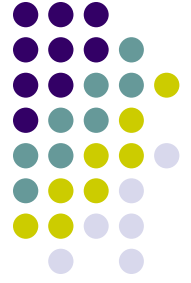
- Zayıf bir varlık kümesinde, niteliklerin değerleri ile varlıkları birbirinden ayırdetmek mümkün değildir.
- Zayıf bir varlık kümesinin anlamlı olabilmesi için şu özelliklere sahip olması gerekir:
  - Bu varlık kümesi ile güçlü bir varlık kümesi arasında birden-bire ya da (güçlüden-zayıfa) birden-çoğa bir ilişki bulunmalıdır.
  - Zayıf varlıklar için bu ilişkinin var olma bağımlılığı oluşturmmalıdır.
  - Zayıf varlık kümesinin nitelikleri arasında, aynı güçlü varlığa bağlı zayıf varlıkları birbirinden ayırt etmeyi sağlayan bir nitelik grubu (discriminator) bulunmalıdır.
- Böylece zayıf bir varlığın anahtarı, bağlı olduğu üstün varlığın anahtarına ayırıcı nitelikler eklenerek elde edilir.

# Zayıf Varlığı Güçlendirme Örneği



- Türkiye'deki tüm lise öğrencilerinin bilgilerini içeren ÖĞRENCİ varlık kümesi zayıf bir varlık kümesidir. Çünkü farklı liselerde öğrenci numarası, adı ve soyadı aynı olan öğrenciler bulunabilir.
- ÖĞRENCİ varlık kümesi ile LİSE varlık kümesi arasında bir OKUYAN ilişkisi kurulursa, öğrencileri birbirinden ayırdetmek için kullanılan OGR\_NO niteliğine, LİSE varlık kümesinin anahtarı olan LİSE\_KODU eklenir. Bu durumda ÖĞRENCİ varlık kümesinin anahtarı (LİSE\_KODU, OGR\_NO) ikilisi olur.

# Varlık-İlişki Çizelgeleri



Varlık Kümesi

Nitelik

Zayıf Varlık  
Kümesi

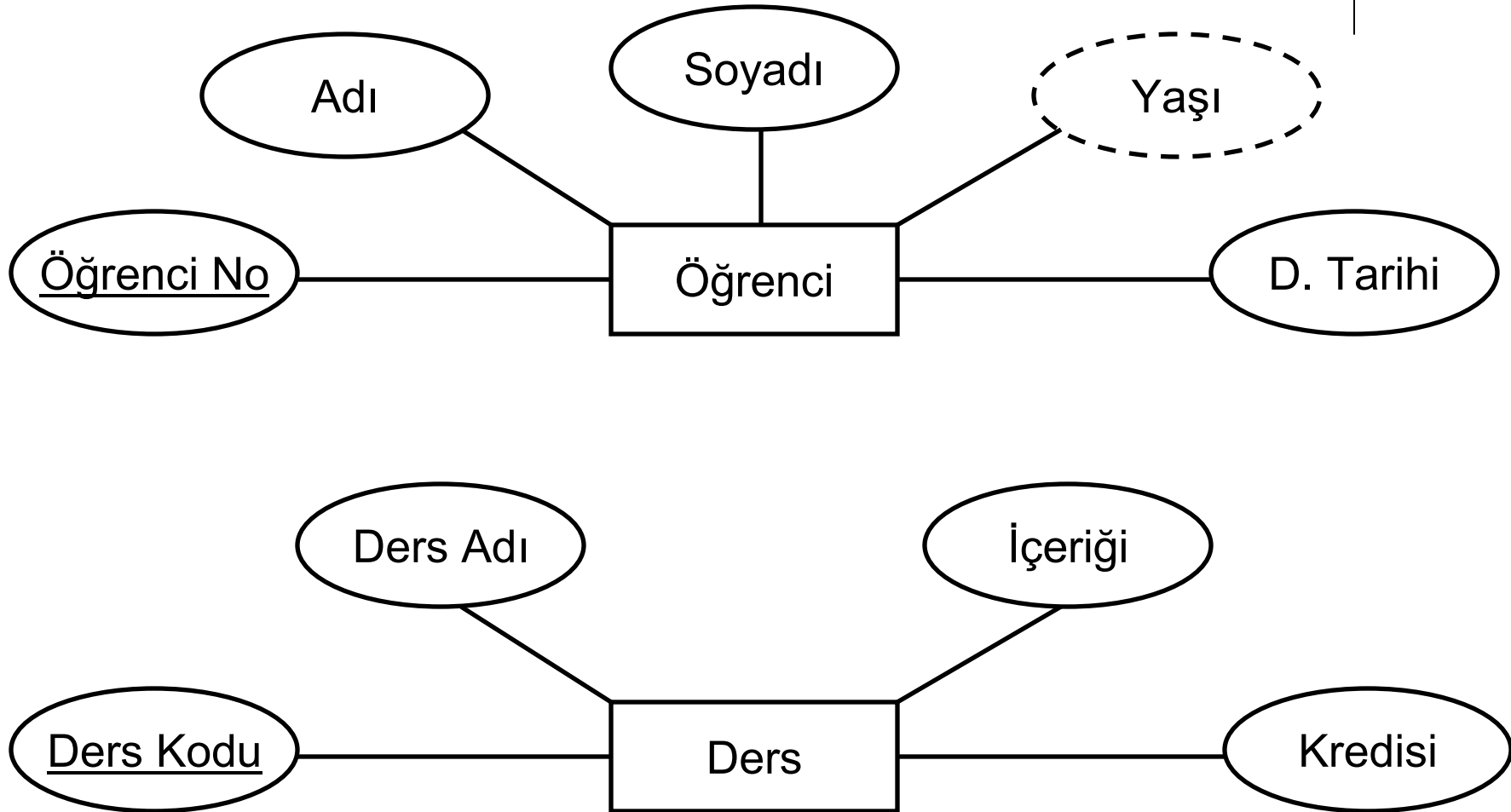
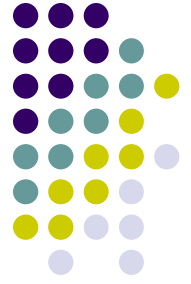
Anahtar Nitelik

İlişki Kümesi

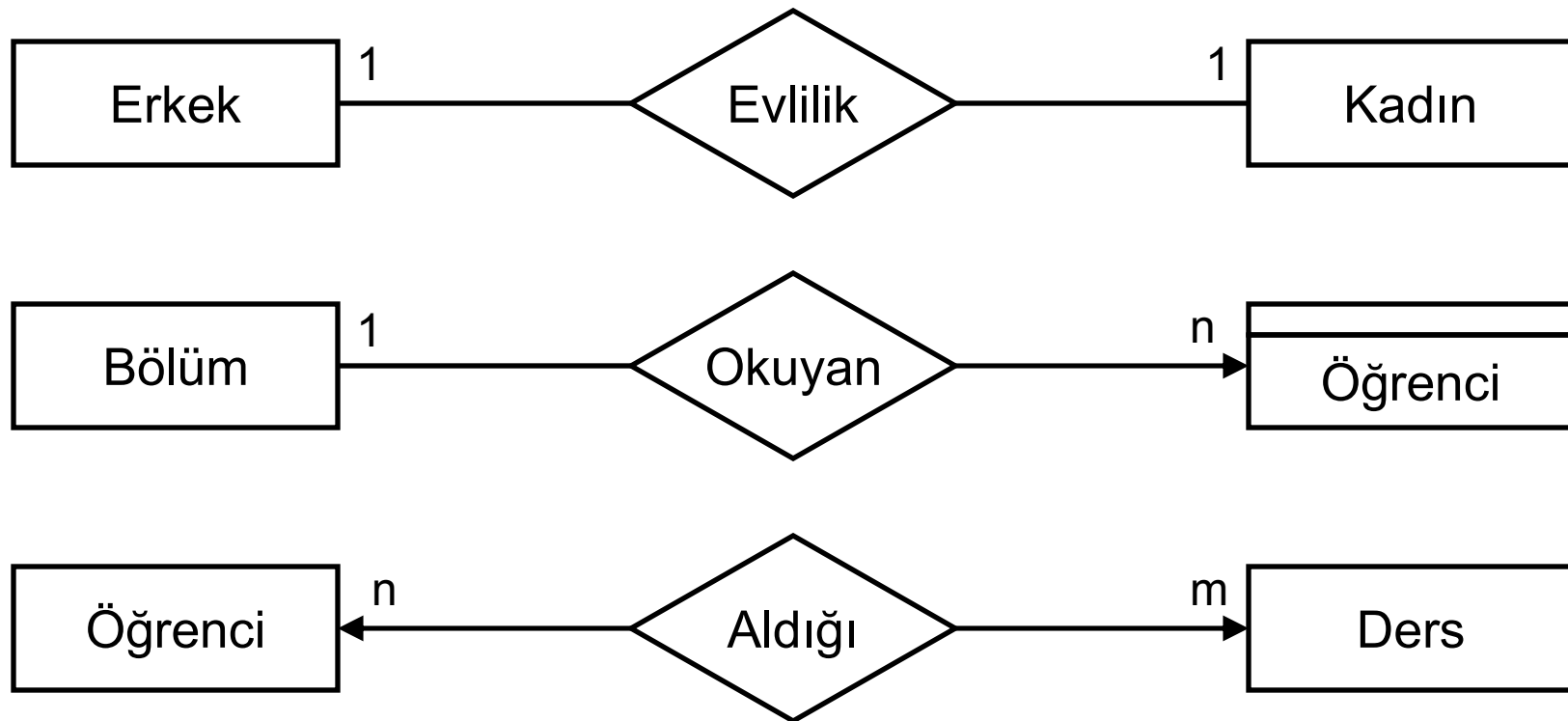
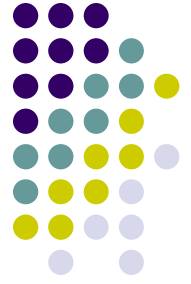
Bağımlı Varlık  
Kümesi

Türetilen Nitelik

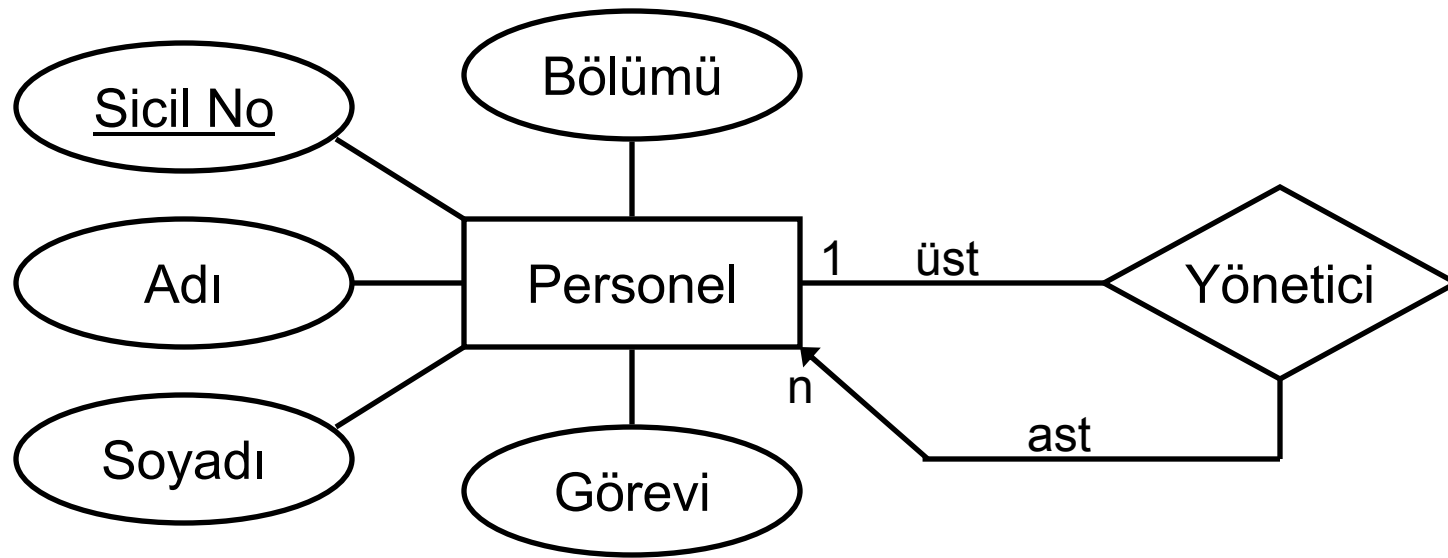
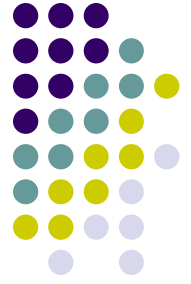
# Varlık Kümesi ve Nitelik Örnekleri



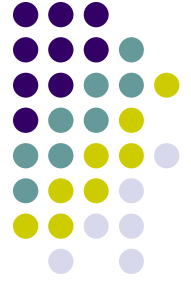
# İlişki Kümesi Örnekleri



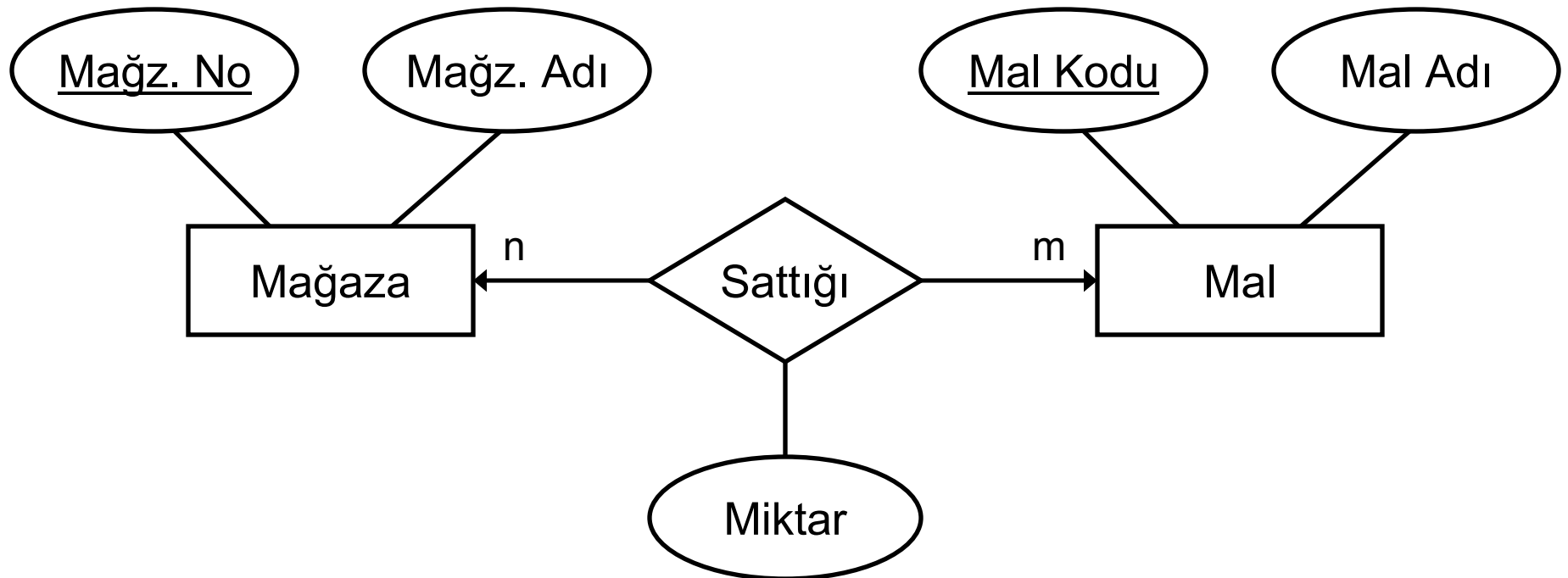
# Rol Örnekleri



# İlişkilerde Nitelik



- İlişkilerde de tanımlayıcı nitelikler bulunabilir.
- Aşağıdaki “Miktar” niteliği “Sattığı” ilişkisi için tanımlayıcı niteliklerdir.



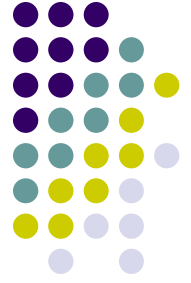
# Varlık, İlişki ve Nitelik seçimleri



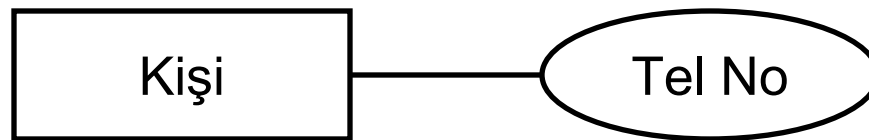
- Varlık-ilişki modelinin sunduğu 3 temel kavram: varlık kümesi, nitelik, ilişki kümesi kavramlarıdır.
- Gerçek dünyayı soyutlayarak semantik modelleme yapmak için kullanılabilen bu üç kavram birbirinden kesin çizgilerle ayrılmamaktadır.
- Varlık kümesi ile nitelik, ya da varlık kümesi ile ilişki kümeleri birbirinin yerine kullanılabilir; aynı konuda değişik düzenlemeler yapılabilir.



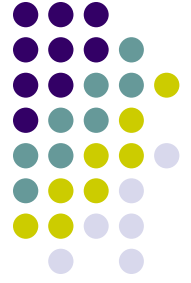
# Örnek : Telefon Numarası (1)



- “telefon numarası” kişi varlık kümesinin bir niteliği olarak düşünülürse;
  - “telefon numarası” kişilerden bağımsız olarak varolamaz.
  - bir kişinin sadece bir telefon numarası bulunabilir.
  - birden çok kişinin telefon numarası aynı olabilir (telefon numarası kişi varlık kümesinin anahtarlarından biri olarak tanımlanmadığı sürece).



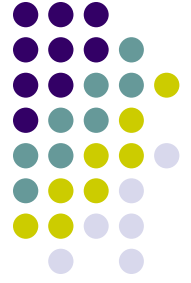
# Örnek : Telefon Numarası (2)



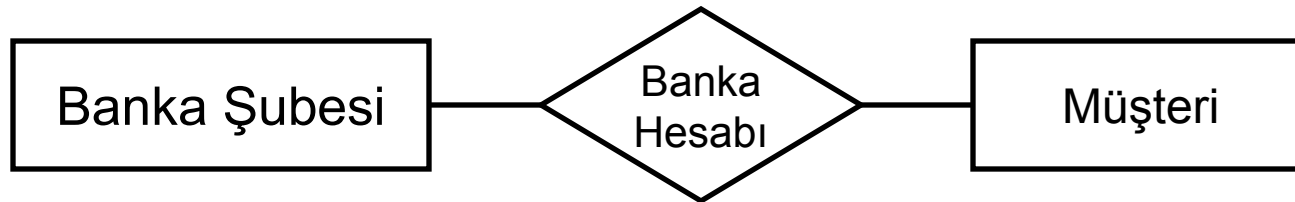
- “telefon numarası” ayrı bir varlık kümesi olarak düşünülüp, bu varlık kümesi ile kişi varlık kümesi arasında ilişki kurulursa;
  - telefonun numarası dışında nitelikleri de bulunabilir.
  - kişi ve telefon varlık kümeleri arasındaki ilişkinin türüne göre her kişinin bir ya da birçok telefonu olabilir.
  - bir telefon numarası bir ya da birçok kişiye verilebilir.



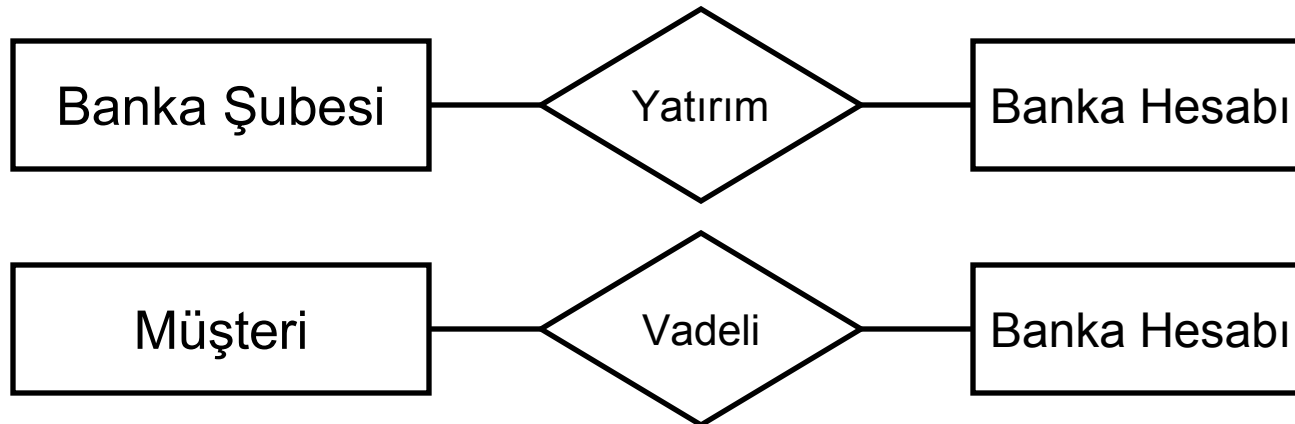
# Örnek : Banka Hesabı



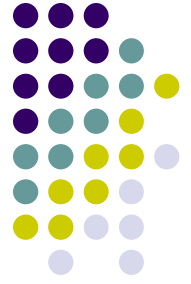
- "Banka hesabı" banka şubesi ve müşteri varlık kümeleri arasında bir ilişki olarak düşünülebilir.



- "Banka hesabı" ayrı bir varlık kümesi olarak düşünülüp bu varlık kümesi ile banka şubesi ve müşteri varlık kümeleri arasında birer ilişki de kurulabilir.



# Seerken Dikkat !



- Veri modellemede varlık kümelerinin, niteliklerin ve ilişki kümelerinin seçimi çok önemlidir, Ancak bunların nasıl seçileceğine ilişkin kesin kurallar da yoktur.
- Kuruluşun öncelikleri ve uygulamaların özellikleri yanında veri modellemeyi gerçekleştiren bilişim teknik personelinin anlayışı da düzenlemede etkili olmaktadır.