

# Varlık-İlişki Modeli (Entity-Relationship Model)

- 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 (Entity-Relationship Model)

- 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 – İlişki Modelinin Bileşenleri

- Varlık
- Nitelik
- İlişki

# Varlık – İlişki Modelinin Bileşenleri

- Varlık Nedir?
  - Var olan ve benzerlerinden ayırt edilebilen her nesneye **varlık** (entity) denir.
  - E-R Diyagramda **Dikdörtgen** şekli ile temsil edilir.

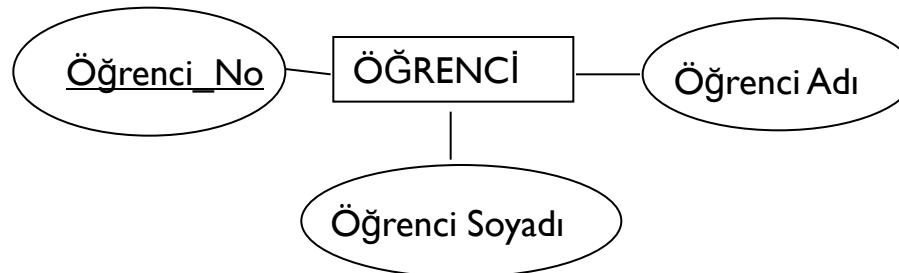
Örnek:

ÖĞRENCİ

# Varlık – İlişki Modelinin Bileşenleri

- Nitelik Nedir?
  - Bir varlık kümesindeki varlıkların özelliklerini göstermek ve varlıkları birbirinden ayırt etmek için nitelikler (attributes) kullanılır.
  - E-R Diyagramda **Oval / Elips** şekli ile temsil edilir.
  - 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.

**Örnek:**



# Nitelik Çeşitleri

- Tek- Değerli Nitelik (Single-Valued Attribute)
  - Bir varlığın bir niteliğinin aldığı değer tek ise bu niteliğe tek değerli nitelik denir.

Örnek:

ÖĞRENCİ Numarası → “080987”

# Nitelik Çeşitleri

- Çok- Değerli Nitelik (Multi-Valued Attribute)
  - Birden fazla değer alabilen niteliklere çok-değerli nitelik denir.

## Örnek:

Bir ÖĞRENCİ'nin birden fazla telefon numarası veya e-posta adresi olabilir.

# Nitelik Çeşitleri

- Birleşik Nitelik (Composite Attribute)
  - Birden fazla nitelik birleştirilerek yeni bir nitelik oluşturulabilir. Bu tür niteliklere birleşik nitelik adı verilir.

## Örnek:

“mahalle”, “cadde”, “sokak”, “apartman”, “posta kodu” ve “şehir” gibi nitelikler birleştirilerek “adres” isimli yeni bir nitelik oluşturulabilir.



# Nitelik Çeşitleri

- Türetilen Nitelik (Derived / Calculated Attribute)
  - Bir nitelik kullanılarak bir başka varlık niteliği elde edilebiliyorsa, bu yeni niteliğe türetilen nitelik adı verilir.

## Örnek:

“personel” varlığının “doğum tarihi” niteliğinden yararlanılarak “yaş” niteliği elde edilebilir. Bu örnekte “yaş” niteliği türetilen nitelik, tasarımda ayrıca tanımlanmasına gerek yoktur.

# Varlık – İlişki Modelinin Bileşenleri (Devam)

## İlişki Nedir?

- Varlıklar arasındaki bağıntıya ilişki adı verilir.
- E-R Diyagramda **Baklava** şekli ile temsil edilir.

## Örnek:

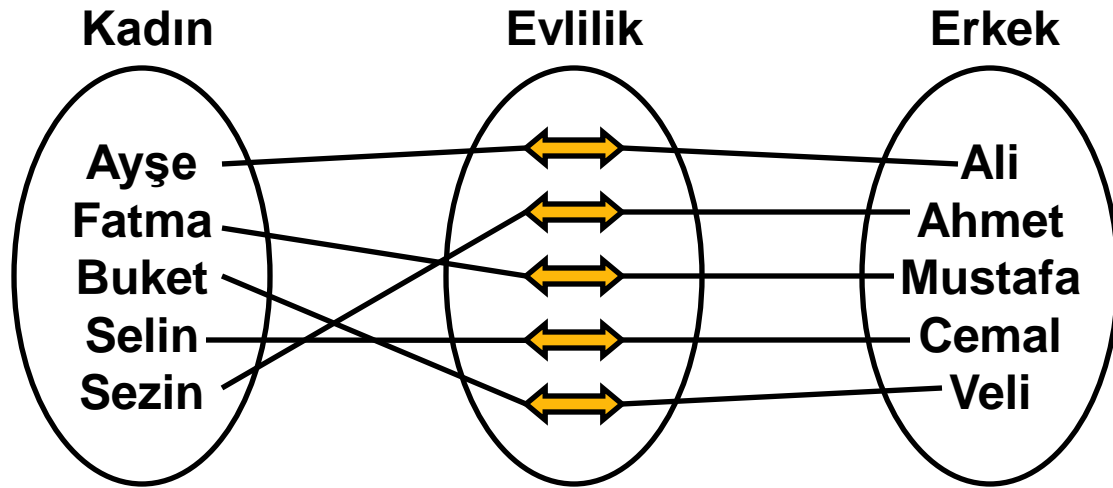


# İlişki Türleri

- 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)

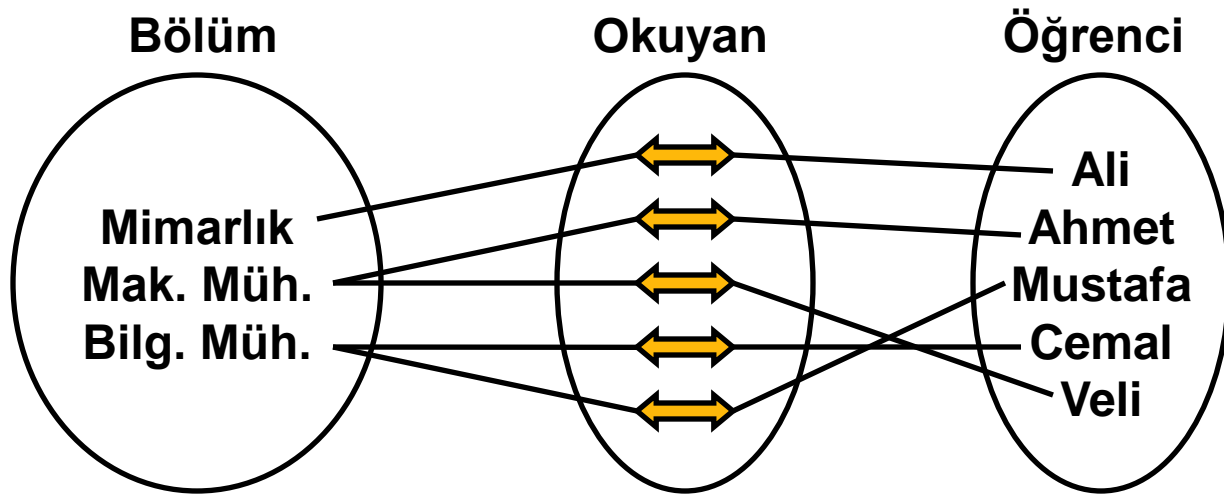
# İlişki Türleri : Birden-bire (1-1)

- 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.



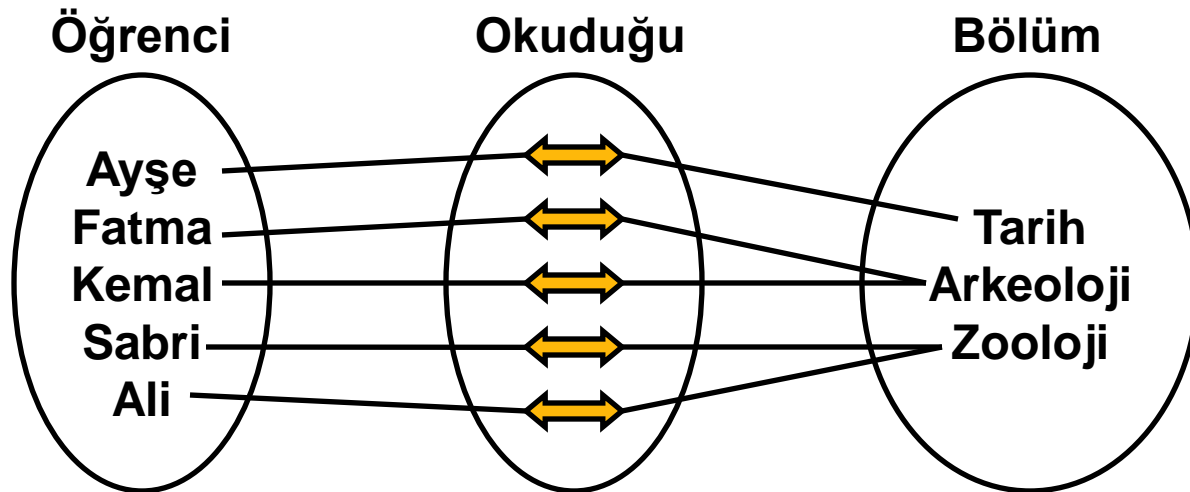
# İlişki Türleri : Birden-çoğa (1-n)

- 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.



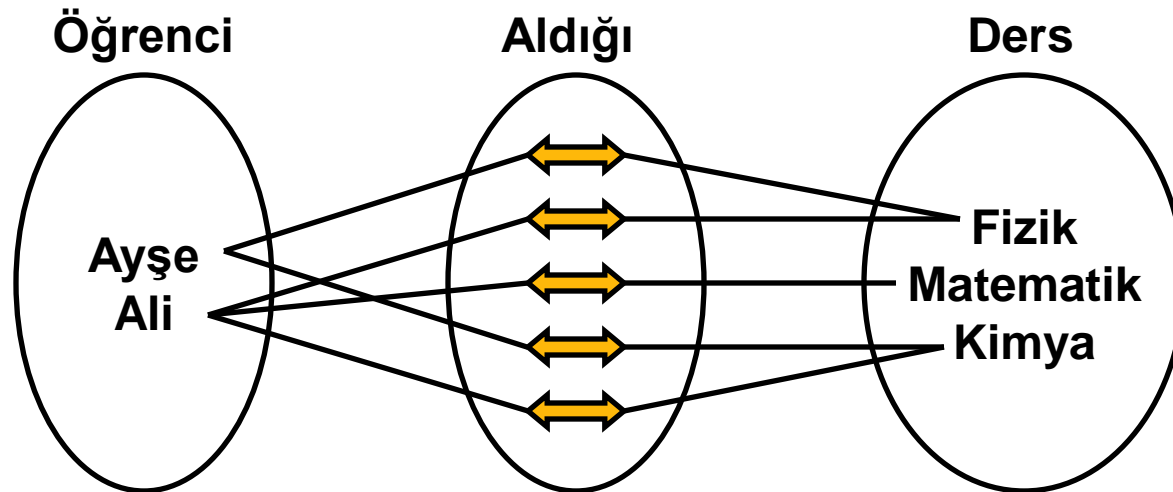
# İlişki Türleri : Çoktan-bire (n-1)

- 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 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.



# Rol

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

# Rol

- Oysa kişi<sub>1</sub> ve kişi<sub>2</sub> arasında kurulan evlilik ilişkisinde kişilerden hangisinin erkek hangisinin kadın olduğunun belirtilmesi gerekebilir (bazı isimler hem erkek hem kadınlar tarafından kullanıldığı için).
- Benzer biçimde bir kurumda çalışan personel<sub>1</sub> ve personel<sub>2</sub> arasında kurulan yönetici ilişkisinde hangi personelin rolünün yönetici (üst), hangi personelin rolünün ise yönetilen (ast) olduğunun belirtilmesine gerek vardır.

# 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)

# Anahtar Nitelik

- 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, varlık kümeleri için kullanılır.
  - Her varlık için bir nitelik anahtar olarak seçilmelidir.
  - Anahtar varlıkları belirtmek için altını çizmemiz yeterlidir.

# Varlık-İlişki Çizelgeleri

Varlık Kümesi

Nitelik

İlişki Kümesi

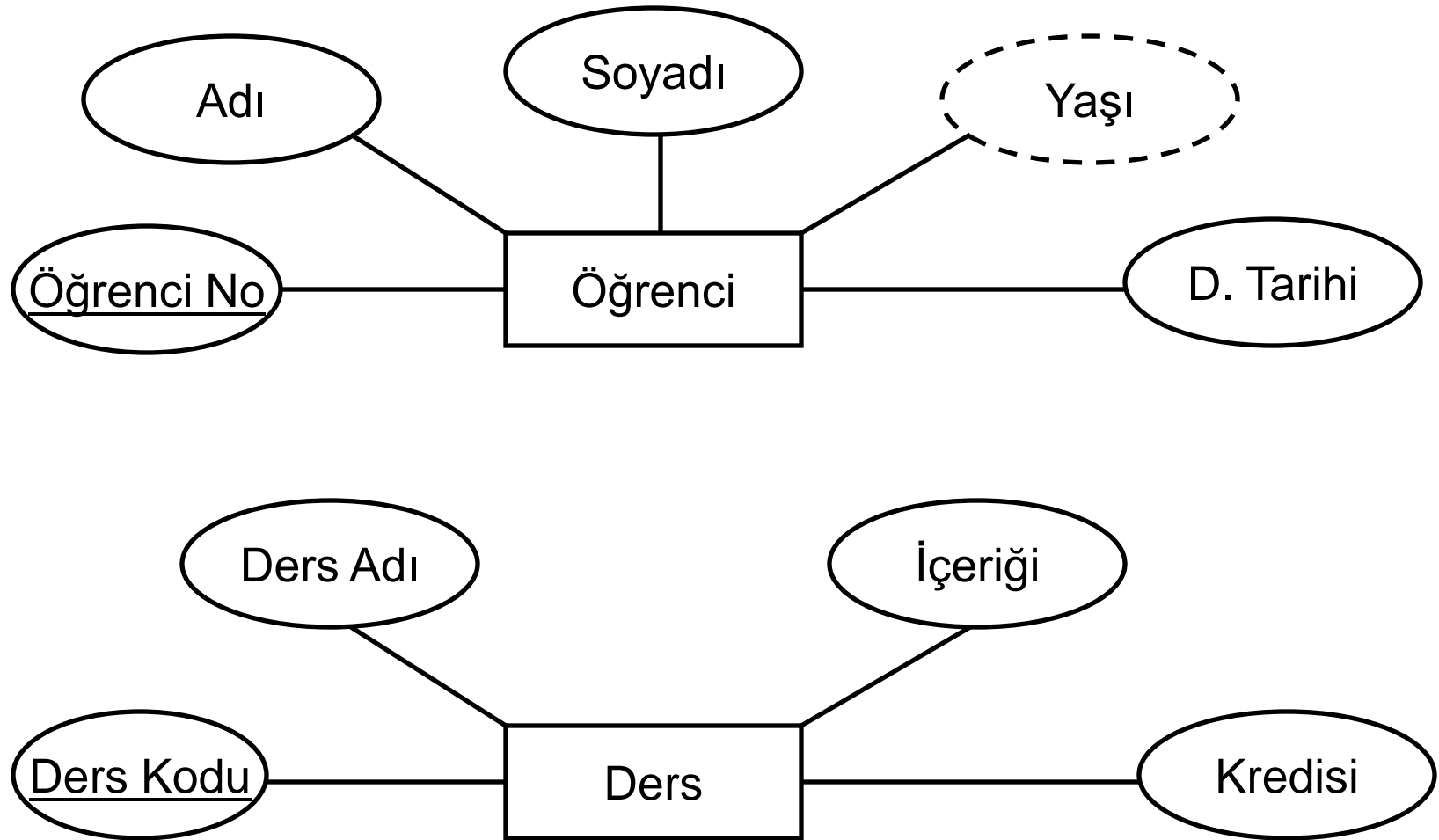
Anahtar Nitelik

Türetilen Nitelik

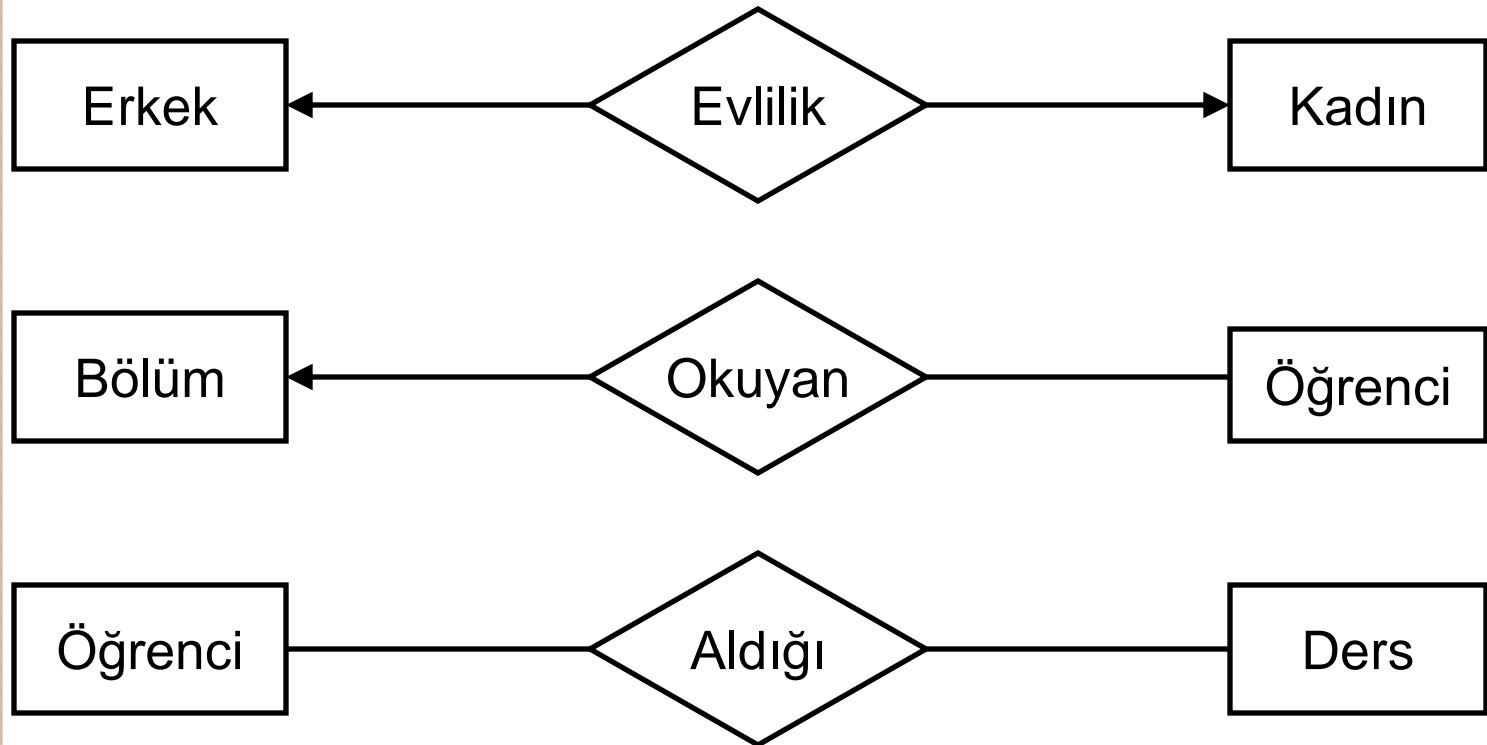
Çok-Değerli Nitelik

Birleşik Nitelik

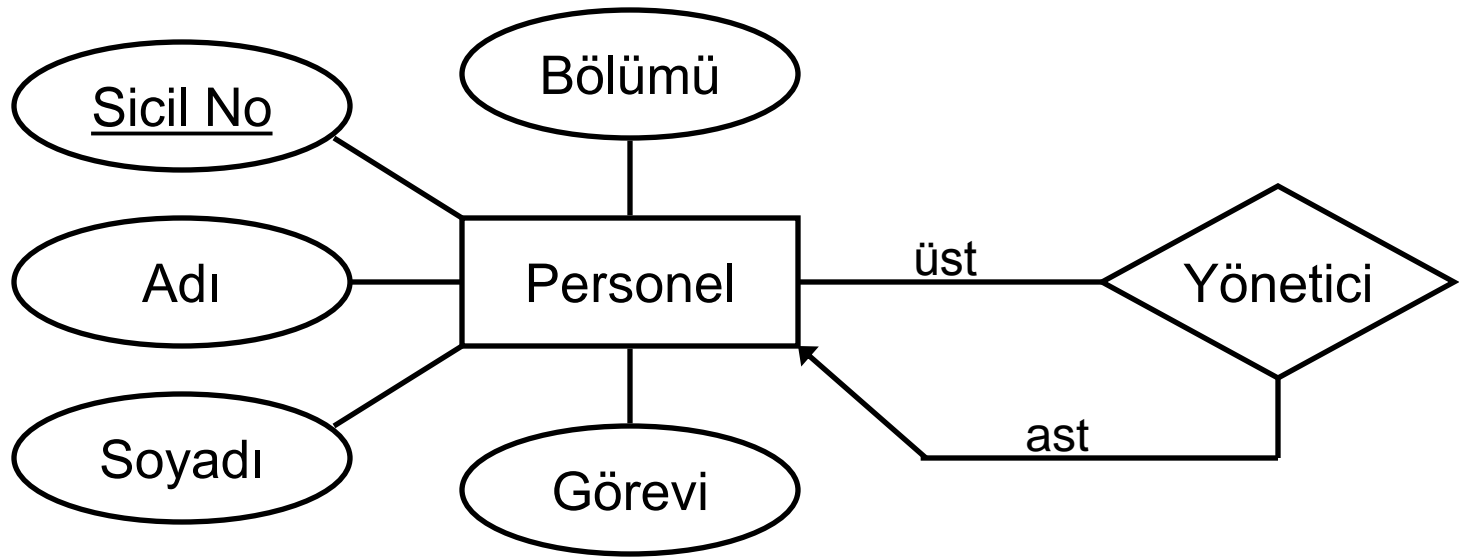
# Varlık Kümesi ve Nitelik Örnekleri



# İlişki Kümesi Örnekleri



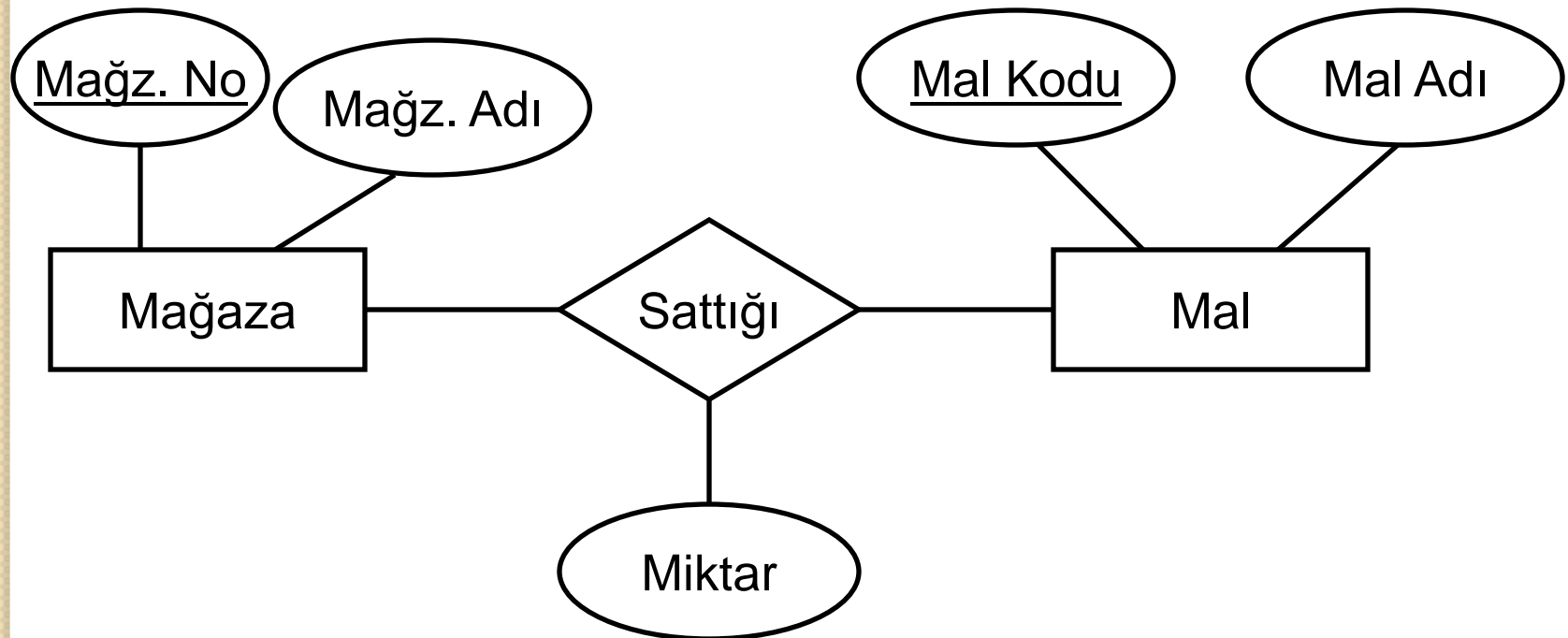
# Tekrarlamalı (Recursive) İlişki





# İlişkilerde Nitelik

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



# Alıştırma I

- Bir futbol oyuncusu sadece bir takımda oynayabilir, ancak bir futbol takımında birden fazla oyuncu vardır.
- Yukarıdaki açıklamaya göre Varlık-İlişki şemasını çiziniz.

## Alıştırma 2

- Bir müşteri birden fazla otomobile sahip olabilir ve her otomobile birden fazla müşteri sahip olabilir.
- Bu ilişkinin varlık-ilişki şemasını çiziniz.

# Alıştırma 3

- Veritabanı Gereksinimleri
  - Şirket, BÖLÜM'lerden oluşmaktadır. Her bölümün bir numarası, bir adı ve bir yöneticisi vardır.
  - Her BÖLÜM, belli sayıda PROJE kontrol etmektedir. Her projenin bir numarası, bir adı, bir bütçesi ve bir süresi vardır.
  - Veritabanında şirket ÇALIŞAN'larının sicil numarası, adı, maaşı, cinsiyeti ve doğum tarihi saklanacaktır. Her çalışan bir BÖLÜM'de yer almaktadır, fakat birden fazla PROJE'de çalışabilir. Her çalışanın, projelerde haftalık çalışma saatleri kaydedilecektir. Her çalışan bir üst amirinin kim olduğu tutulacaktır.

# Çalışma Soruları

- I. Bir şirketin şubeleri ve çalışanlarının tutulacağı veri tabanının Varlık-İlişki diyagramını çiziniz:
- Bir şirketin birden fazla şubesi vardır. Her şubenin birden fazla çalışanı vardır, ve çalışanların numarası, adı, soyadı, aldığı maaş, adresi, ve görevi veri tabanında saklanmalıdır. Her çalışan, sadece bir şubede çalışabilir ancak bir şubede birden fazla çalışan olabilir. Bunun yanısıra her şubede çalışan elemanlardan biri yöneticilik yapmaktadır ve bir çalışan sadece bir şubeyi yönetebilir.
  - Not: Her varlığın birincil anahtarını gösteriniz.

# Çalışma Soruları

2. Veritabanı gereksinimleri aşağıda verilmiştir. Buna göre Varlık-İlişki Diyagramını çiziniz.

Film Kiralama Şirketi için Veritabanında Film, Film Yönetmeni ve Sistemin Üyeleri hakkında veriler saklanacaktır. Her üyenin, kullanıcıAdı, şifresi ve adresi sistemde tutulmalıdır. Üyelerin kullanıcıAdı sistemde sadece bir tek üye tarafından kullanılabilir. Filmlerin FilmNo'su, başlığı, çıkışYılı ve kategorisi de veri tabanında saklanacaktır. Filmlerin FilmNo'su belirleyicidir. Her filmin sadece bir yönetmeni vardır, ancak her yönetmen birden fazla filmin yönetmenliğini yapabilir. Her yönetmenin, yönetmenNo'su, adı, soyadı, ve adresi veritabanında saklanmalıdır. Yönetmenlerin adresi, cadde, şehir ve ülke verilerinden oluşan birleşik niteliklidir. Bir film, birden fazla üye tarafından kiralanabilir, ve her üye de birden fazla film kiralayabilir. Yukarıda belirtilen ilişkiye göre hangi üyenin ne zaman hangi filmi kiraladığı/iade ettiği bilgisin de ihtiyaç olacağından, kiralamaTarihi ve İadeTarihi de veritabanında saklanmalıdır.

# Çalışma Soruları

3. Bir Sanal Alışveriş Sitesi için geliştirilecek sistemin gereksinimleri aşağıda verilmiştir.

Her Firma için FirmaNo'su (belirleyici), Adresi ve Telefonu saklanacaktır. Her firma bir çok ürün tedarik eder, ve bir ürün de sadece bir firma tarafından tedarik edilir. Ürünlerin ürünNo'su, ürünAdı, SatışFiyatı ve AlışFiyatı veritabanında saklanmalıdır. Ayrıca sistemde her ürünün ait olduğu kategori bilgisine de ihtiyaç vardır. Her ürünün ait olduğu sadece bir kategori vardır, ancak her kategoride birden fazla ürün bulunabilir. KategoriAdı, KDVOoranı, KategorID (belirleyici) bilgileri de her kategori için saklanmalıdır. Sanal Alışveriş Sitesinde Müşteri bilgileri de saklanacaktır. Her müşterinin MüşteriNo'su (belirleyici), adı, soyadı, adresi ve telefonu vardır. Ayrıca müşteri siparişleri de sistemde tutulacaktır. Müşteriler, birden fazla sipariş verebilirler, ancak bir sipariş sadece bir müşteri tarafından verilebilir. Her siparişte birden fazla ürün yer alabilir, bir ürün de birden fazla siparişte bulunabilir. Ürün siparişlerinde ayrıca hangi üründen kaç adet sipariş edildiği de saklanmalıdır. Ayrıca siparişler için sipariş tarihleri de saklanmalıdır.

# Çalışma Soruları (Devam)

## 4. Ev Kiralama şirketi için Varlık-İlişki şemasını çiziniz.

Sistemde ev, kiracı ve ev sahibi bilgileri saklanacaktır. Her ev için ev no'su, ev tipi, oda sayısı, kira ücreti veritabanında saklanacaktır. Bir evin bir veya birden fazla sahibi olabilir ,bir ev sahibinin de bir veya birden fazla evi olabilir. Ev sahibinin kimlik no'su, adı, soyadı , telefon numaraları ve cinsiyeti veritabanında saklanmalıdır. Bir kiracı (bir kişi veya aile olabilir) bir ev kiralayabilir, bir ev ise sadece bir kişi (kiracı) tarafından kiralanabilir. Kiracılar için kimlik no'su, adı, soyadı, telefonu, cinsiyeti ve medeni hali veri tabanında tutulmalıdır. Kiracıların evleri ne zaman kiraladıkları (kira sözleşmesinin başlangıç ve bitiş tarihleri ) bilgisinin de veritabanında tutulması gerekmektedir.

Not: Her varlığın birincil anahtarını gösteriniz.