

-152116025- TASARIM SÜREÇLERİ

Ders 7: Tasarım Evresi-2

Dr. Yıldıray Anagün,

yanagun@ogu.edu.tr

Eskişehir Osmangazi University

Computer Engineering Department

Tasarım Evresinde Neler Yapılır?

- ☐ Kullanıcı-Sistem ara yüzü tasarımı
 - ☐ **Veri tabanı tasarımı**
 - ☐ Gömülü Donanım tasarımı
 - ☐ Yazılım tasarımı
 - ☐ Sistem kontrolleri/test tasarımı.
 - ☐ Ağ tasarımı ve güvenliği
-

Veri Tabanı

- Veri tabanı birbirleriyle ilgili olan bilgi ve dosyaların bir arada, belli bir düzen içinde tutulduğu bir sistemdir.



Veri Tabanı

- ☐ Düz Dosyalar
- ☐ Word
- ☐ Note pad
- ☐ Excel
- ☐ xml
- ☐ Her Türlü Kayıt



Veri Tabanı

- ❑ Böylece, düz metin dosyasında, ya da Excel dosyasında tuttuğunuz değerler ya da veriler bütünü de bir çeşit veritabanıdır.
 - ❑ Ancak, bu şekilde bir veritabanının modern ve ideal düzgünlükte tutulmuş olması beklenemez.
 - ❑ Bu nedenle Veri tabanı Yönetim Sistemi (VTYS) genel adı ile birtakım yazılımlar kullanarak veritabanı yaratmak daha uygun bir yaklaşımdır.
-

Veri Tabanı Yönetim Sistemleri

- ☐ MS SQL Server
 - ☐ Oracle
 - ☐ My SQL
 - ☐ IBM DB2
 - ☐ Microsoft Access
 - ☐ **Progress**
 - ☐ SQLite
 - ☐ **Teradata**
 - ☐ **CSQL**
 - ☐ **OpenLink Virtuoso**
 - ☐ FileMaker
 - ☐ Firebird
 - ☐ MongoDB, Hadop(No-SQL)
-

Veri Tabanı Yönetim Sistemleri Faydaları

- ❑ Veri fazlalığı ve tutarsızlığının önlenmesi
 - ❑ Verilere ulaşma kolaylığı
 - ❑ Tek tip ve formatta dosya kullanımı
 - ❑ Diğer uygulamalarla bütünleşme kolaylığı
 - ❑ Bir seferde güncelleme (Atomicity)
 - ❑ Eş zamanlı erişim
 - ❑ Güvenlik
-

Veri fazlalığı ve tutarsızlığının önlenmesi:

- ❑ Gereksiz veri (redundancy)
 - ❑ Farklı Yazım (inconsistency)
 - ❑ VTYS herhangi bir veriyi kaydettiğinde onu tek bir yere kaydederek indeksleme yardımıyla o veriyi tekrar tekrar kullanır.
-

Verilere ulaşma kolaylığı:

- ☐ İndeksleme yöntemi
 - ☐ Arama
 - ☐ Sıraya Dizme

 - ☐ İndeksleme yardımıyla veriler hızlı bir şekilde aranır ve ulaşılır.
-

Tek tip ve formatta dosya kullanımı:

- ☐ Farklı kişiler farklı dosyalar
 - ☐ Farklı programlar farklı dosyalar
-

Diğer uygulamalarla entegrasyon kolaylığı:

- ☐ Farklı VTYS
 - ☐ MERNIS
 - ☐ MOBESE
 - ☐ YOK
 - ☐ Diğer
-

Bir seferde güncelleme (Atomicity):

- Aynı anda iki ayrı kaynaktaki değişimin yönetilmesi.
 - Örnek:
 - A hesabı B hesabına 100 TL gönderiyor. Bu arada A hesabı bakiyesi 125 TL, B hesabı bakiyesi 50 TL.
 - İşlem sırasıyla şu şekilde yapılacaktır.
 1. $A = 125 \text{ TL} ; B = 50 \text{ TL}$
 2. $A = A - 100$
 1. $A = 25 \text{ TL}$
 3. $B = B + 100$
 1. $B = 150 \text{ TL}$
 - Bu işlemler sırasında tam 2 numaralı işlem yapılırken sistem kapansa ne olur?
 - A 25 TL ye inmiş ancak B hala 50 TL de kalmış olacaktır.
-

Eş Zamanlı Erişim:

- ❑ Birden fazla kişinin aynı anda işlem yapması
 - ❑ Örnek
 - ❑ Hesabına para gönderirken eş zamanlı ve bir birlerinden bağımsız olarak C hesabı da B hesabına 50 TL göndesin. İşlemin adımları şöyle olacaktır.
 1. $A = 125 \text{ TL}$; $B = 50 \text{ TL}$, $C = 200 \text{ TL}$
 2. $A = A - 100$ (aynı anda) $C = C - 50$
 1. $A = 25$; $C = 150$
 3. $B = B + 100$; $B = B + 50$
 1. $B = 150$
 2. **$B = 100$!**
-

Eş Zamanlı Erişim:

- ❑ Görüldüğü gibi aynı anda iki işlem B üzerinde işlem yapınca B iki defa güncellenmiş ve B hesabına toplam $100 + 50 = 150$ TL transfer Yapılmasına rağmen B hesabına geçen sadece 50 TL olmuştur.
 - ❑ Bu arada 100 TL kaybolmuştur. Bir telefon şirketinin son ödeme günü hesap hareketlerini düşünürseniz, aynı anda birden fazla kişinin aynı hesaba ulaşmak isteme olasılığının hiç de düşük olmadığını görebilirsiniz.
-

Güvenlik:

- ☐ Program İzalasyonu
- ☐ Kullanıcı İzalasyonu
- ☐ Kriptolama
- ☐ Log Tutma
- ☐ Kullanım hakkı tanıma



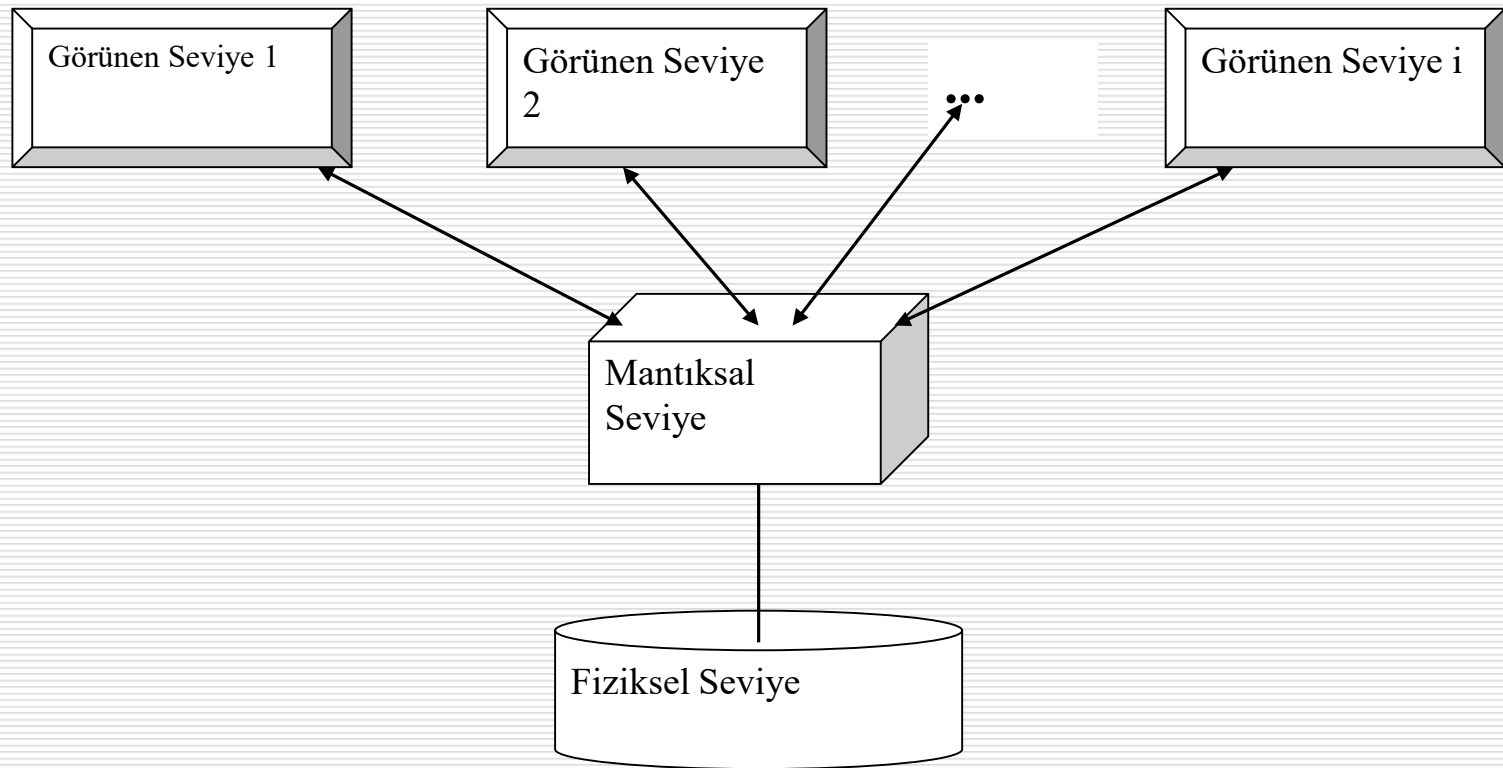
VTYS sistemi katmanları

- Fiziksel Katman
 - En alttaki seviyedir, karmaşık veri yapılarının bulunduğu, verilerin nerede, nasıl tutulduğunun belirlendiği seviyedir.
 - Mantıksal Katman
 - Hangi verilerin tutulacak
 - Veriler arasındaki ilişki
 - Veri tabanı yöneticisinin, müdahale ettiği seviyedir.
 - SQL Komutları
 - Tablo Yapıları
 - Görünen Katman
-

Görünen Katman

- ❑ Kullanıcılar, veri tabanının sadece küçük bir kısmıyla ilgilenirler.
 - ❑ Sistem içinde bir çok Görünen seviye olabilir. Uygulama programları ve arayüzler kullanılarak bu seviye sağlanır
-

Görünen Katman



Veri Modelleri

- Veri tabanları oluşturulurken, verilerin tanımlanması, verileri arasındaki ilişkinin belirlenmesi, mantıksal yapının oluşturulması aşamalarında kullanılan yöntem ve tekniklerin bir araya gelmesiyle üç ayrı tipte veri modeli oluşmuştur .Bunlar,
-

Veri Modelleri

- ❑ Nesne Temelli Mantıksal Modeller
 - Varlık İlişkisel Model
 - Nesneye Yönelik Model
 - Semantik Model
 - İşlevsel Veri Modeli
 - ❑ Kayıt Temelli Mantıksal Modeller
 - İlişkisel Model
 - Ağ (Network) Modeli
 - Hiyerarşik Model
 - ❑ Fiziksel Modellerdir
 - Birleştiren Model (unifying model)
 - Frame - Bellek Modelidir
-

Veri Tabanı Şemaları

- ❑ Mantıksal Katmanda Veri tabanının nasıl kurulduğunu gösteren çizimsel yapılardır.
 - ❑ Tablolar (Varlıklar ve İlişkiler)
 - ❑ Özellikler (attribute) ya da alan isimleri (fields)
-

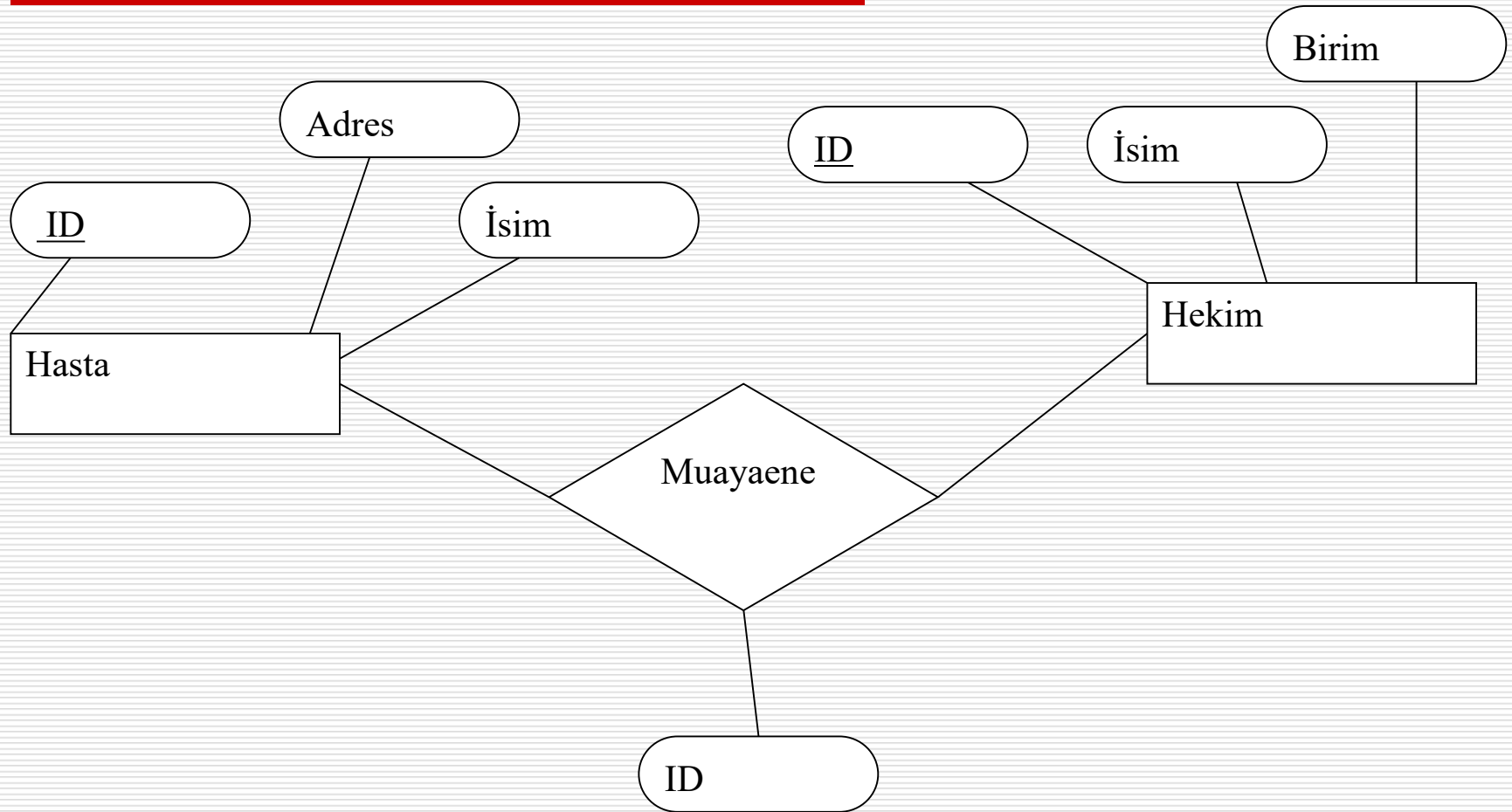
Veri Tabanı Şemalarında Bulunması Gereken Açıklamalar

- ❑ Her bir tablonun ne amaçla kullanılacağı
 - ❑ Tablodaki alan isimlerinin değişken tipleri (string, varchar, integer vs.)
 - ❑ Her bir alanın boş olup olamayacağı
 - ❑ Her bir tablo üzerinde bulunan ya da bulunması istenen *trigger* – tetikleyiciler- ile ilgili açıklamalar
 - ❑ Gerekli *stored procedure* açıklamaları
-

Şemalarda Kullanılan Simgeler

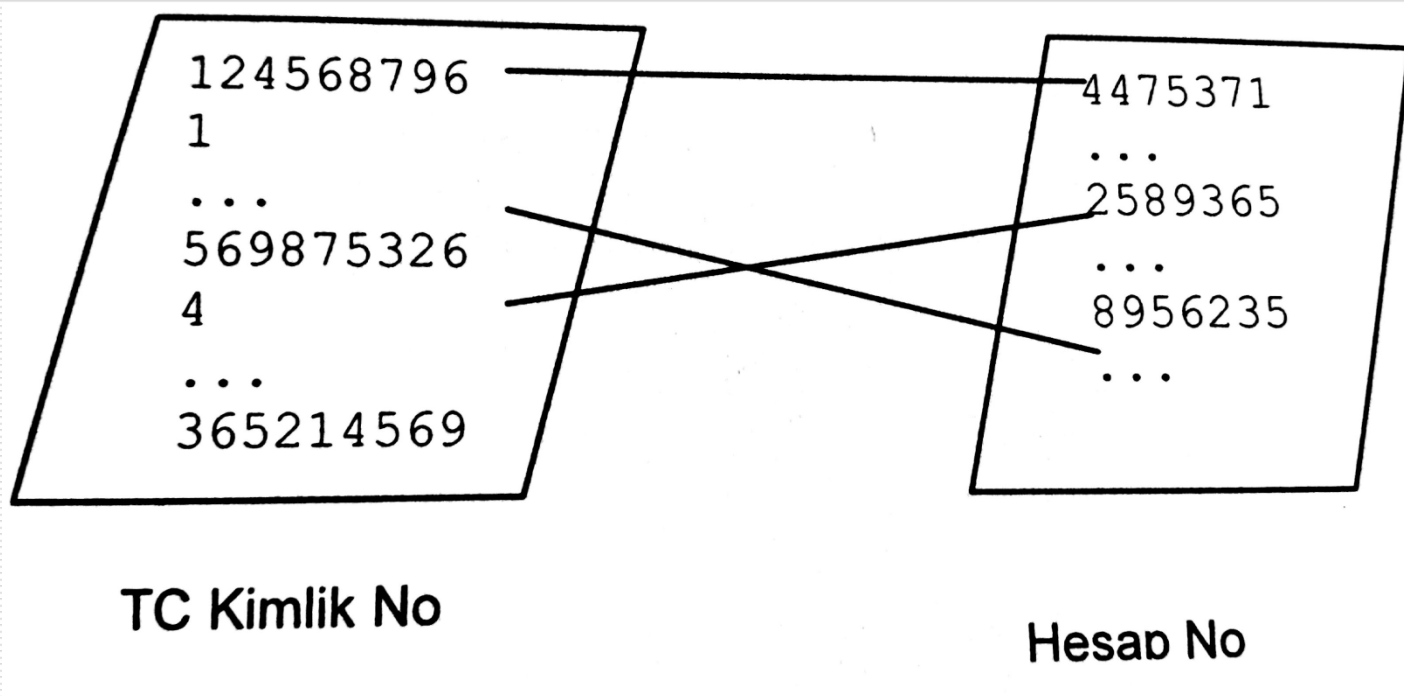


Örnek Şema



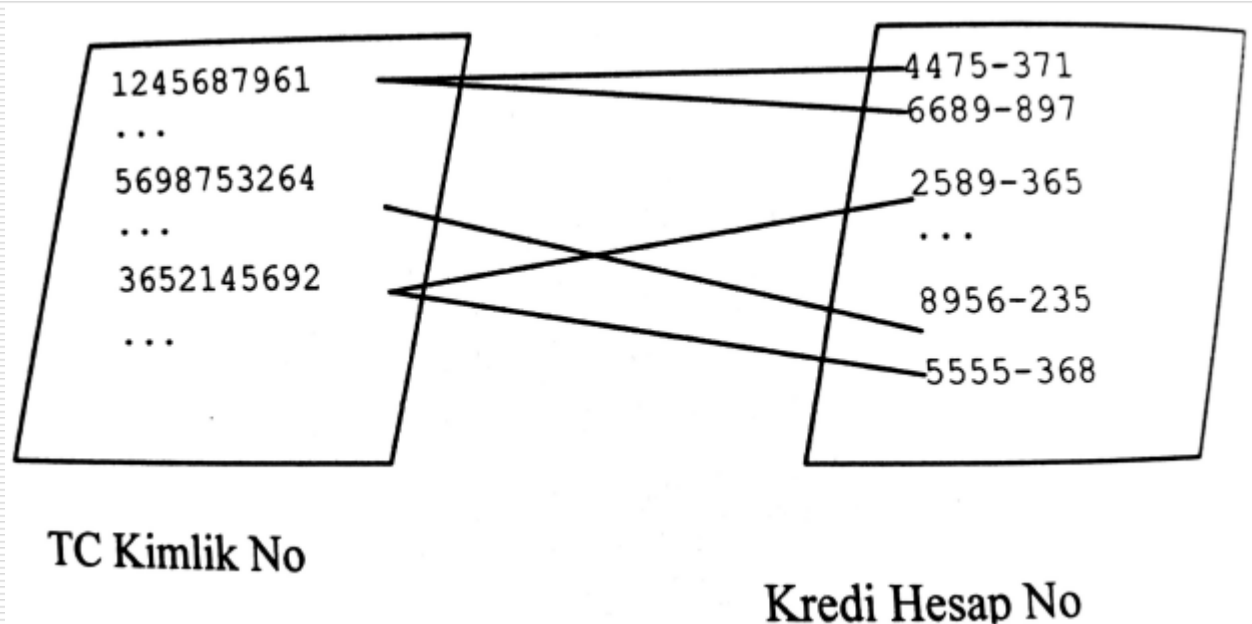
Varlıklar arası ilişki boyutları

□ Bire Bir (One to One)



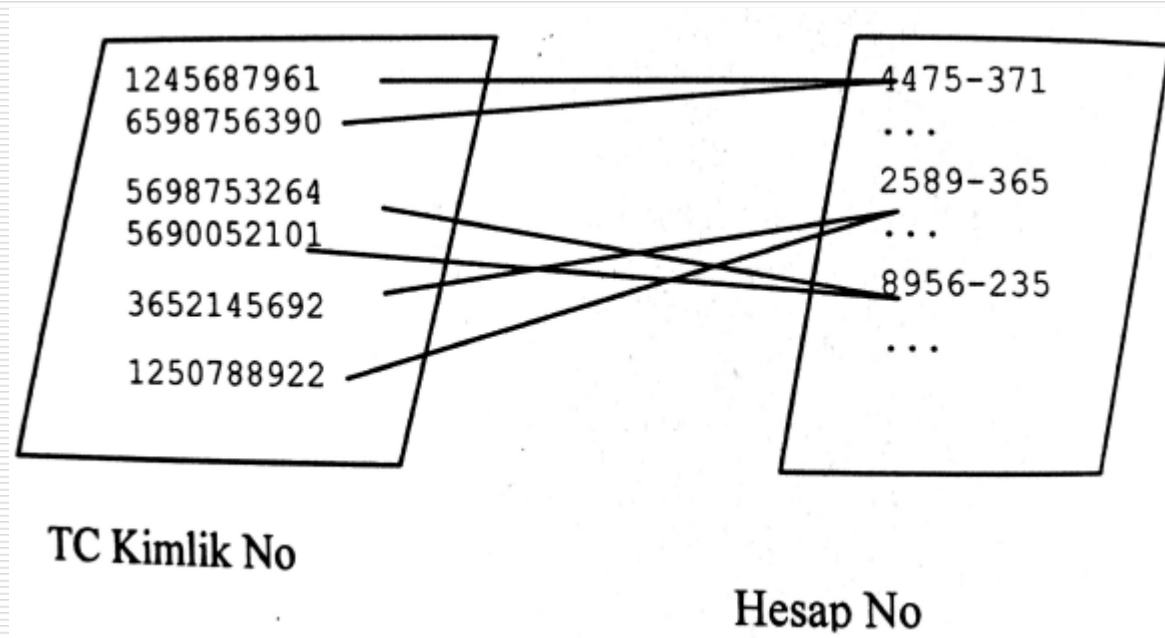
Varlıklar arası ilişki boyutları

□ Bire Çok (One to Many)



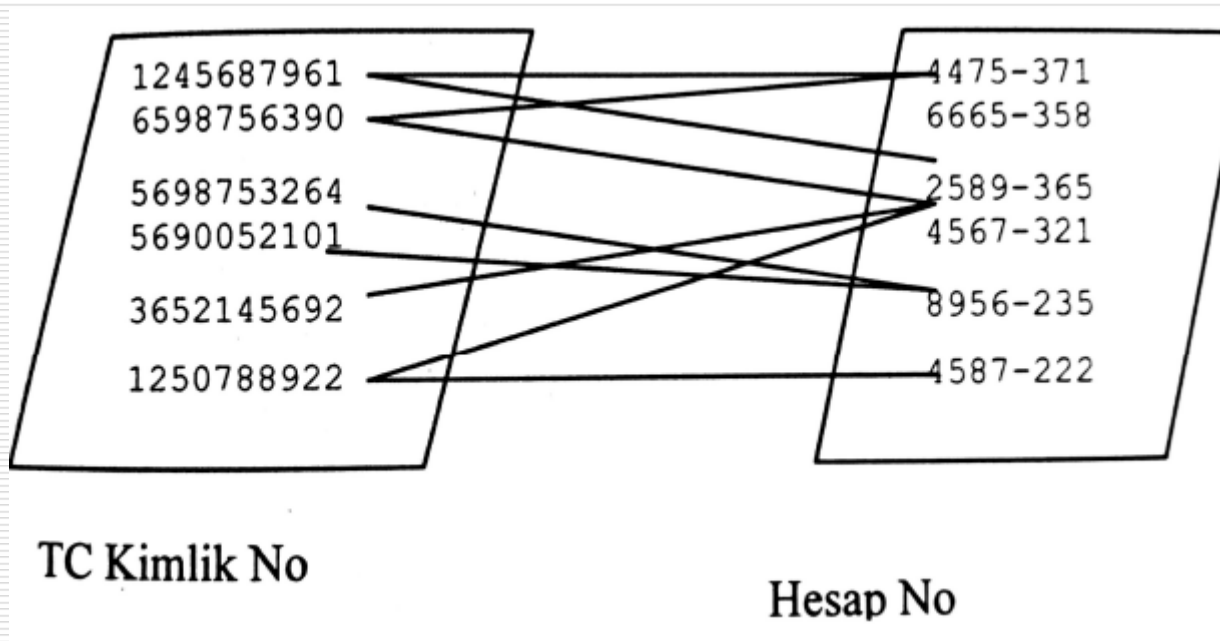
Varlıklar arası ilişki boyutları

❑ Çok-a Bir (Many to One)



Varlıklar arası ilişki boyutları

□ Çok-a Çok (Many to Many)



Uygulama

- ❑ Bir tıp merkezi için veri tabanı tasarlayınız.
 - ❑ Tıp merkezi çeşitli branşlarda, ayaktan tedavi hizmeti vermektedir.
 - ❑ Kendi bünyesinde laboratuvar hizmeti de sunmaktadır.
-

Uygulama (Devam)

- ❑ SGK ile işbirliği halinde olduğundan reçeteleri eczanelerde işlem görmektedir.
 - ❑ Reçete – ilaçlar- veritabanına kaydedilmelidir.
 - ❑ Kaba VT çizildikten sonra bir sonraki yansının gereklerini yerine getiriniz
-

Uygulama (Devam)

- ❑ Tıp merkezi bilişim sisteminde bir yükseltme uygulamasına girmiştir: Bu uygulama çerçevesinde, aşağıdaki durumlarda kesinlikle kullanılmaması gereken *bazı* ilaçlar için hekim yazılım tarafından uyarılacaktır.
 - ❑ Uygulama yapılabilmesi için veritabanında ne gibi bir değişiklik yapabilirsiniz.
-

Uygulama (Devam)

- Durumlar.
 - Tansiyon Diastolik < 6 sistolik >15, şeker < 10 veya > 150, PLT (Platelets): , MCHC: < 20 Mean Corpuscular Hemoglobin Concentration , MCH:< 15 (Eritrositlerdeki hemoglobin miktarını gösterir.) Eosinophils: (Eozinofiller) > 35 (Allerjide ve paraziter hastalıklarda yükselir.)
 - Hemotokrit :

| Erkek | Kadın |
|----------------|-------|
| 13-15 yaş >%49 | %46 |
| 16-50 yaş %53 | %46 |
| >50 yaş %55 | %46 |
-

Tasarım Çıktıları

- ❑ Veritabanı isterler dökümanı
 - ❑ E/R Diyagramı
 - ❑ Veri Tabanı Tabloları
 - ❑ Stored Procedures, Triggerrs, vb. sorgular, Constraints ...
 - ❑ Veritabanı Güvenliği(Kendisi, alış-veriş; text tabanı veriler, multimedya)
-