

Veritabanı genellikle bir bilgisayar sisteminde elektronik olarak depolanan yapılandırılmış bilgi veya veriden oluşan düzenli bir koleksiyondur. Veritabanı genellikle bir [veritabanı yönetim sistemi (DBMS)](https://www.oracle.com/tr/database/what-is-database/#WhatIsDBMS) ile kontrol edilir. Veri ve DBMS ve aynı zamanda bunlarla ilişkili uygulama yazılımları bir araya getirildiğinde veritabanı sistemi olarak ifade edilir.

* 1960 – Charles Bachman ilk DBMS sistemini tasarladı.
* 1970 – Codd tanıtıldı IBM'Bilgi Yönetim Sistemi (EYS).
* 1976 – Peter Chen, ER modeli olarak da bilinen Varlık-ilişki modelini icat etti ve tanımladı.
* 1980 - [ilişkisel model](https://www.guru99.com/tr/relational-data-model-dbms.html) yaygın olarak kabul edilen bir veritabanı bileşeni haline gelir.
* 1985 – Nesneye yönelik DBMS geliştirildi.
* 1990 – Nesne yöneliminin ilişkisel DBMS'ye dahil edilmesi.
* 1991 - Microsoft MS erişimini, kişisel bir DBMS'yi sunar ve bu, diğer tüm kişisel DBMS ürünlerinin yerini alır.
* 1995 – İlk internet veritabanı uygulamaları.
* 1997 – Veritabanı işlemeye XML uygulandı. Birçok satıcı XML'i DBMS ürünlerine entegre etmeye başlar

Veri tabanı, **birbirleriyle ilişkili bilgilerin depolandığı alana** verilen addır. (database)

veritabanı, veriyi koruma, veriler arasında ilişki kurma, farklı kullanıcılara farklı yetki sınırları içinde veriye erişim imkânı sunma işlevlerini de yerine getirir.

Veri Olguların, kavramların veya talimatların insan tarafından veya otomatik yolla iletişim, yorumlama ve işleme amacına uygun bir biçimde ifadesidir. Genellikle, biz veri veya veri birimleri üzerindeki işlemlerimizi varlık hakkında her hangi bilgi almak için gerçekleştiririz. Veri kaydedilebilir bilinen gerçeklerdir. Örneğin bir kişinin ismi, adresi, telefon numarası gibi.

Veritabanı Yönetim Sistemi (VTYS-Database Management System); veritabanı tanımlamak, veritabanı oluşturmak, veritabanında işlem yapmak, veritabanının farklı kullanıcı yetkilerini belirlemek, veritabanının bakımını ve yedeklemesini yapmak için geliştirilmiş programlar bütünüdür.

Veritabanı ve veritabanı yönetim sisteminin birlikte oluşturduğu bütün ise veritabanı sistemi olarak ifade edilir.

Veritabanı Tanımlamak: Veritabanında yer alacak verinin adı, tipi, uzunluğu gibi veri yapıları ve özelliklerinin belirlenmesidir.

Veritabanı Oluşturmak: Veritabanını yaratma ve veriyi depolama sürecidir. Verilerin depolanacağı fiziksel bellek alanının belirlenmesi ve verilerin bu alana aktarılmasını içerir.

Veritabanında İşlem Yapmak: Veritabanını izleme, veriler üzerinde sorgulama yapma, veritabanında gerekli olan değişiklikleri yaparak veritabanını güncelleme ve verilerden rapor elde etme işlevlerini içerir.

Veriye Erişim

1. Sıralı erişim : istenilen veriye ulaşılıncaya kadar ilgili dosyadaki tüm verilerin sırayla okunması gerekir.
2. Doğrudan Erişim : verilerin yer aldığı fiziksel adresler birer indeks numarası ile tanımlanıp bu indeks numaraları da ayrı bir dosya olarak saklanır. Herhangi bir veriye erişilmek istendiğinde, önce indeks dosyasından verinin yer aldığı adresi gösteren indeks numarası bulunur, daha sonra bu numaraya karşılık gelen fiziksel adrese doğrudan erişim sağlanır.

Veritabanı yaklaşımından önce kullanılan ve geleneksel dosya sistemi olarak ifade edilen yaklaşımda veriler bilgisayarda ayrı ayrı dosyalar biçiminde saklanmaktaydı. Birbiriyle ilişkili olan veriler bir dosyada, başka bir açıdan birbiriyle ilişkili veriler de başka bir dosyada yer almaktaydı. Bununla birlikte iki farklı dosya içinde aynı verinin yer alması da söz konusuydu.

BÖLÜM--->KULLANICILAR--->UYGULAMA PROGRAMI 1---> TÜRETİLMİŞ DOSYALAR (A-B-C-D...)

BÖLÜM2--->KULLANICILAR--->UYGULAMA PROGRAMI2--->TÜRETİLMİŞ DOSYALAR2(C-O-S-T)

geleneksel dosya sisteminde ilgili kuruluşun her alt sistemi kendi faaliyetlerini gerçekleştirmek için o alt sisteme özgü uygulamalar ve veri dosyalarına ihtiyaç duyar.

OLUMSUZLUKLARI........

• Veri tekrarı ve veri tutarsızlığına yol açar. • Veri paylaşımına olanak vermez. • Uygulamalarda ihtiyaç duyulan değişikliklerin gerçekleştirilebilmesi için uzmanlık bilgisi gerektirir. • İstenilen veriye ulaşmada güçlükler bulunur. • Verilerin güvenliği ve gizliliği konusunda sorun yaşanır. • Veriler ve uygulamalarla ilgili belirli bir standart yoktur. •Verileri yedekleme ve kurtarma konusunda güçlükler yaşanır

VERİTABANININ ÜSTÜNLÜĞÜ..........

•Veri tekrarı ve veri tutarsızlığınI önler

• Veri paylaşımına olanak verir. • Uzmanlık bilgisine ihtiyaç duyulmayacak derecede kullanım kolaylığı sağlar • İhtiyaç duyulan veriye, tanımlanmış kullanıcı yetkileri kapsamında kolaylıkla erişilmesini sağlar. •Veri güvenliği ve gizliliğini güçlü bir biçimde yerine getirir. • Veriler ve uygulamalarla ilgili standart yapı ve kuralların olması kullanım kolaylığı sağlar. • Verileri yedekleme ve kurtarma konusunda kolaylık sağlayan programlar barındırır. VTYS’ler, sağladıkları avantajlara rağmen maliyet açısından geleneksel dosya sistem

VERİ TABANI KULLANICILARI

1. Veritabanı sorumluları :, veritabanının tasarlanması, oluşturulması ve veritabanının işletim faaliyetlerinden birinci derecede sorumlu olan ve veritabanı üzerinde en fazla yetkiye sahip olan kullanıcılardır. Veritabanı sorumluları, veritabanı yöneticisi ve veritabanı tasarımcısı olarak iki başlık altında incelenebilir.

* Veritabanı yöneticisi : Veritabanı yöneticisinin (database administrator) veritabanına erişim yetkilerini belirleme, veritabanı kullanımının düzenlenmesi ve izlenmesini sağlama, ihtiyaç duyulan yazılım ve donanım kaynaklarını edinme biçiminde sıralanan sorumlulukları vardır. Ayrıca güvenlik ihlalleri ve kötü sistem yanıt süresi gibi sorunların çözümünden de sorumludur.
* Veritabanı tasarımcısı : anı Tasarımcısı Veritabanı tasarımcısı (database designer) veritabanında saklanacak olan verilerin tanımlanmasından ve bu verilerin depolanması ve gösterilmesi için gerekli olan uygun yapıların seçilmesinden sorumludur.
* Veritabanı tasarımını yapma: Veritabanının farklı düzey şemalarının oluşturulması, veriler üzerinde yapılacak her türlü işlem için gereksinimlerin belirlenmesi, veritabanı içeriğinin oluşturulması.
* Bütünlük kısıtlamalarını belirleyip tanımlama: Veritabanında veri bütünlüğünün sağlanabilmesi, veri kaybının önüne geçilebilmesi, veri bütünlüğünü tehlikeye sokacak kullanıcı hatalarının önlenmesi amacıyla gerekli kurallar, ilişkiler ve kısıtlamaların belirlenmesi.
* Veritabanı kullanım yetkilerini tanımlama: Veritabanı kullanıcılarının ve kullanım yetkilerinin tanımlanması ile her kullanıcı grubunun hangi veriler üzerinde hangi işlemleri yapmaya yetkili olduğunun belirlenmesi. Bu tanımlamalar sonrasında ilgili kullanıcılara yetkili oldukları erişimin sağlanması
* Veritabanı güvenliğini sağlama: Kullanıcıdan ya da yazılım sorunlarından kaynaklanabilecek veri kaybının önlenmesi amacıyla veritabanının yedeklerinin alınması ve kurtarma işlemlerine ilişkin düzenlemelerin yapılması
* Veritabanının işletimini izleme ve sürekliliğini sağlama: Veritabanın kullanıma sürekli açık olmasının sağlanması ve sistemde meydana gelebilecek herhangi bir sorunu hızlı biçimde giderebilecek tedbirlerin alınması.
* Güncelleme ihtiyaçlarına cevap verebilme: Kullanıcı ihtiyaçlarında ortaya çıkabilecek değişikliklerin izlenebilmesi ve bu değişikliklere paralel olarak veritabanı içeriği, şema tanımları, bütünlük kısıtlamaları, fiziksel yapı ile ilgili parametreler, kullanıcı tanımları ve kullanıcı yetkilerinde gerekli değişikliklerin oluşturulması ve tanımlanması
* Veritabanından beklenen performansı sağlama: Veritabanı kullanıcılarının beklentilerine cevap verecek bir yapının donanım ve yazılım ihtiyaçlarının sağlanması, veritabanındaki tanımlamaların, kısıtlamaların kullanıcı ihtiyaçlarına cevap verebilecek yeterlilikte olup olmadığının izlenmesi, gerekiyorsa değişikliklerin yapılması.

1. Son kullanıcılar : on kullanıcılar (end users), yaptıkları işler gereği veritabanına sorgulama ya da güncelleme yapmak veya rapor türetmek için erişen kullanıcılardır. Bu tür kullanıcılar veritabanı ile kullanıcının bağlantısını sağlayan ve uygulama programcıları tarafından geliştirilen yazılımları kullanırlar.

* Standart son kullanıcılar : Veritabanına nadiren erişim yapan fakat her seferinde farklı bilgi ihtiyacı olabilen kullanıcılardır. Bu tür kullanıcılar isteklerini belirtmek için gelişmiş veritabanı sorgu dili kullanırlar
* Sıradan ya da Parametik son kullanıcılar :. Bu kullanıcıların temel iş fonksiyonları, veritabanı üzerinde sürekli bir sorgulama ve güncelleme yapmalarını gerektirir. Standart (önceden belirlenmiş) sorgu ve güncelleme yaparlar.
* Gelişmiş son kullanıcılar : VTYS’nin sağladığı özellikler yardımıyla ayrıntılı olarak belirledikleri karmaşık gereksinimlerini karşılamak amacıyla veritabanını kullanan gruptur.
* Bağımsız son kulllanıcılar : Bu kullanıcılar menü kullanımı ya da araç çubukları gibi grafiksel ögeler yardımıyla kullanım kolaylığı sağlayan hazır paket programlarını kullanarak kişisel veritabanlarının sürekliliğini sağlar.

SİSTEM ANALİSTLERİ VEUYGULAMA PROGRAMCILARI

Sistem analisti son kullanıcıların, özellikle de sıradan son kullanıcıların gereksinimlerini belirleyen ve standart işlemler yoluyla bu gereksinimleri karşılayabilecek ayrıntıları belirleyen kişi ya da kişilerdir. Uygulama programcıları ise sistem analisti tarafından belirlenen ayrıntıları program hâline getiren ve daha sonra test eden, hataları ayıklayan, belgeleyen ve kaydedilmiş işlemler olarak sürekliliğini sağlayan kişilerdir.

VERİ TABANI TÖNETİM SİSTEMLERİNİN MİMARİSİ

Basit bir istemci/sunucu VTYS mimarisinde sistem fonksiyonel olarak iki modüle ayrılmaktadır. İstemci modülü (client modül), VTYS’nin herhangi bir kullanıcı iş istasyonunda ya da kişisel bilgisayar üzerinde çalışan parçasıdır. Tipik olarak istemci modülünde veritabanına erişmek için uygulama programları ve kullanıcı arayüzleri bulunur. Sunucu modülünde (server modüle) ise veri deposu ile veri deposuna erişimi ve sorgulamayı sağlayacak fonksiyonlar yer alır.

VERİ MODELLERİ

Veri soyutlama (data abstraction), verilerin düzenlenmesi ve depolanmasına ilişkin ayrıntıların gizlenmesi ve verinin daha iyi anlaşılmasını sağlamak için veriye ilişkin temel özelliklerin vurgulanması anlamına gelir. Veri modeli ise söz konusu bu soyutlamaları gerçekleştirebilmek için gerekli olan araçları sağlar. Veri modeli (data model), bir veritabanının mantıksal yapısını tanımlamada kullanılacak kavramlar, işlemler ve kurallar bütünüdür. Veritabanının mantıksal yapısı; veri tipleri, veriler arasındaki ilişkiler, veri üzerinde uygulanacak kısıtlamalar vb. dir. Veri modellerinin çoğu verinin geri çağrılması ve veritabanı üzerinde güncelleme yapmaya yönelik işlemleri de içerir.

VERİ MODELLERİNİN SINIFLANDIRILMASI

Yüksek düzeyli ya da kavramsal veri modelleri (conceptual data models), kullanıcıların veri algılama biçimiyle ilişkili kavramları kapsar. Düşük düzeyli ya da fiziksel veri modelleri (physical data models), verinin bilgisayar ortamında nasıl depolanacağına ilişkin ayrıntıları tanımlayan kavramları kapsar. temsili modeller ya da uygulama veri modelleri (implementation data model) hem son kullanıcılar tarafından kolaylıkla anlaşılabilen kavramları hem de verinin bilgisayarda depolanması yöntemlerine ilişkin kavramları içerir Kavramsal veri modelleri; varlıklar, öznitelikler, ilişkiler gibi kavramları kullanır. Uygulama veri modelleri geleneksel ticari VTYS’lerinde en sık kullanılan modellerdir. Bu modeller geçmişte yaygın olarak kullanılan ağ ve hiyerarşik veri modeli ile günümüzde yaygın olarak kullanılan ilişkisel modelleri içerir. Nesneye yönelik veri modelleri için bir standart ODGM (Object Data Management Group-Nesneye Yönelik Veri Yönetim Grubu) olarak adlandırılır. Nesneye yönelik veri modelleri aynı zamanda yüksek seviyeli kavramsal model olarak da kullanılmaktadır.

Veritabanının herhangi bir veri modeliyle tanımlanması veritabanı şeması (database schema) olarak adlandırılır.

Şema görünümleri şema diyagramı (schema diagram) olarak adlandırılır

Şema diyagramları şemaların yalnızca bazı yönlerini gösterir. Diğer yönler diyagramdan belirlenemez

Veritabanındaki mevcut veriler oldukça sık değişebilir.

r. Bir veritabanının herhangi bir andaki durumuna anlık görüntü denir. Bu aynı zamanda veritabanının mevcut örnek (instance) kümesi olarak da adlandırılır. Veritabanında her şema yapısı kendi oluşum kümesine sahiptir

Veritabanı durumu, belirli bir şema ile ilişkilendirilerek yapılandırılır. Her zaman yeni bir kayıt ekleme, silme ya da bir kayıttaki verinin değerini değiştirmek mümkündür (böylece veritabanının durumu bir durumdan diğerine değiştirilebilir).

ÜÇ ŞEMA MİMARİSİ

1. İçsel (fiziksel) düzey (internal level), veritabanının fiziksel depolama yapısını tanımlayan içsel şemayı içerir. İçsel şema, veriyi depolama ayrıntılarının tamamını ve veritabanına erişim yollarını tanımlayan fiziksel veri modelini kullanır.

2. Kavramsal düzey (conceptual level), kullanıcı topluluğu için tüm veritabanının yapısını tanımlayan kavramsal şemayı içerir. Kavramsal şema fiziksel depolama yapısının ayrıntılarını gizler ve veritabanında yer alan verilerin tipine, veriler arası ilişkilere, kullanıcı işlemlerine ve kısıtlara ilişkin tanımlara yoğunlaşır. Veritabanı sistemlerinde, uygulama veri modelleri kavramsal şemayı tanımlamak için kullanılır. Bu kavramsal şema, yüksek düzeyli kavramsal veri modelleri kullanılarak tasarlanır.

3. Dışsal (görünüm) düzey (external level), bir dizi dışsal şema ya da kullanıcı görünümü içerir. Her dışsal şema bir grup kullanıcının ilgilendiği bazı veritabanı bölümlerini tanımlar. Böylece veritabanının diğer kısmı bu kullanıcı grubundan gizlenir. Her dışsal şema, bir yüksek düzey veri modelinde tasarlanan dışsal şema tabanlı uygulama veri modeli kullanır.

Depolanmış gerçek veri yalnızca içsel (fiziksel) düzeydedir.

Yapılan istek veritabanından bilgi çağırmak ise veritabanından çıkarılan veri, kullanıcının dışsal görünümüyle eşleşecek biçime dönüştürülür. Şema düzeyleri arasında bu şekilde gerçekleştirilen dönüşümler eşleştirme (mapping) olarak adlandırılır.

DIŞSAL DÜZEY--->(DIŞSAL KAVRAMSAL EŞLEŞTİRME) --->KAVRAMSAL DÜZEY--->(KAVRAMSAL İÇSEL EŞLEŞTİRME---->İÇSEL DÜZEY

**DIŞSAL ŞEMA(GÖRÜNÜM)<---->KAVRAMSAK ŞEMA<---->İÇSEL ŞEMA ..DEPOLANMIŞ VERİ..**

Şema düzeyleri arasındaki dönüştürme kapasitesi kısaca veri bağımsızlığı (mantıksal ve fiziksel) olarak ifade edilir. VTYS’lerinde veritabanı tanımları veri tanımlama dili kullanılarak oluşturulur. Veritabanından veri çağırma, veri ekleme, veri silme ve veri üzerinde değişiklikler yapma faaliyetlerini gerçekleştirebilmek amacıyla kullanılan dil veri işleme dili olarak adlandırılırken veritabanından bilgi alma amacıyla sorgulama yapmak için kullanılan veritabanı dili ise sorgulama dili olarak adlandırılır

VERİ TABANI TÜRLERİ

1. Hiyerarşik veritabanı (Hierarchical database) 2. Ağ veritabanı (Network database) 3. İlişkisel veritabanı (Relational database) 4. Nesneye yönelik veritabanı (Object oriented database)

Yaygın olarak kullanılan veritabanı yönetim sistemi yazılımları ise; MS SQL Server, Oracle, MySQL, Sybase, MS Access, PostgreSQL, IBM DB2, Informix, Advantage biçiminde sıralanabilir.