VERİ TABANI 1. ÜNİTE ÖZET

* İşletmeler, faaliyetlerini düzenli ve etkin bir şekilde yürütebilmek için bilgiye ihtiyaç duyarlar.
* Bilgi, günümüzde üretim faktörleri arasında önemli bir yere sahiptir.
* Bilgi sistemleri, işletmelerin zamanında ve amaca uygun bilgiye erişmesini sağlar.
* Veritabanı sistemleri, bilgi sistemlerinin temelini oluşturur ve verilerin toplanması, işlenmesi ve iletilmesini sağlar.
* Veritabanı sistemlerinin tarihçesi, bilgisayar teknolojisinin gelişimiyle paralel olarak evrim geçirmiştir.
* Veritabanı kavramı, veri ve bilgi arasındaki fark, veritabanının özellikleri ve veritabanı yönetim sistemi (VTYS) açıklanmıştır.
* Veriye erişim için sıralı erişim ve doğrudan erişim kavramları ele alınmıştır.
* Geleneksel dosya sistemleri: Veriler ayrı dosyalarda saklanır, ilişkili veriler farklı dosyalara dağıtılır.
* Her alt sistem kendi faaliyetlerini gerçekleştirmek için özgü uygulama ve veri dosyalarına ihtiyaç duyar.
* Veri tekrarı ve tutarsızlığına neden olabilir, veri paylaşımını zorlaştırabilir ve güvenlik konularında sorunlar yaşanabilir.
* Veritabanı yönetim sistemleri: Veri tekrarını ve tutarsızlığını önler, veri paylaşımına olanak sağlar, kullanımı daha kolaydır ve güvenlik konusunda daha güçlüdür.
* Ancak, VTYS'lerin maliyeti yüksektir ve basit uygulamalar için geleneksel dosya sistemleri tercih edilebilir.
* Geleneksel dosya sistemlerinde, her alt sistem için ayrı dosyalar kullanılır, bu da veri tekrarına ve tutarsızlığına yol açar.
* Veritabanı yönetim sistemlerinde ise tüm veriler merkezi bir konumda toplanır, bu da veri tekrarını önler.
* Veri bütünlüğü, verinin değişikliklerinin diğer uygulamalara yansıtılmasıdır ve VTYS'lerde daha kolay sağlanır.
* VTYS'ler aynı anda birden fazla kullanıcıya veri erişimi sağlar, geleneksel dosya sistemlerinden daha esnek ve güçlüdür.
* VTYS'ler kullanıcıların karmaşık teknik bilgilere ihtiyaç duymadan veri işleme yeteneklerine sahiptir.
* Geleneksel dosya sistemleri:
  + Veri tekrarı ve tutarsızlığına yol açar.
  + Veri paylaşımını sınırlar.
  + Uzmanlık gerektirir.
* Veritabanı yönetim sistemleri (VTYS):
  + Gerçek tekr
  + Veri bütünlüğünü sağlar.
  + Veri paylaşımını kolaylaştırır.
  + Kullanımı daha kolay ve esnektir, uzmanlık gerektirmez.

- \*\*Verilerin Gizliliği ve Güvenliği:\*\*

- Kurumlarda ortak kullanılan verilerin depolandığı yapı ve verilerin gizliliği ve güvenliği önemlidir.

- Veri üzerinde yapılabilecek faaliyetler kontrol altında olmalıdır.

- Veritabanı yönetim sistemi, veri yetkilendirmesi ve izleme ile verinin güvenliğini sağlar.

- \*\*Standart Yapı ve Kurallar:\*\*

- Geleneksel dosya sistemlerinde standart olmadığından farklılık ve karmaşıklık görülür.

- Veritabanı sistemlerinde merkezi kontrol ve standart yapılar sayesinde daha düzenli bir yapı oluşturulabilir.

- Standartlar, veri alışverişi ve kullanımını kolaylaştırır.

- \*\*Veritabanı Kullanıcıları:\*\*

- Veritabanı ile etkileşimde olan kişiler farklı rollerde olabilir:

- Veritabanı Sorumluları: Yönetici, tasarımcı ve son kullanıcılar gibi farklı roller vardır.

- Veritabanı Yöneticisi: Yetkilendirme, izleme, güvenlik ve performans gibi görevler üstlenir.

- Veritabanı Tasarımcısı: Veritabanının yapısal tasarımını gerçekleştirir ve kullanıcı ihtiyaçlarını karşılar.

- Temel görevler, veritabanı tasarımı, bütünlük kısıtlamaları, kullanıcı yetkileri, güvenlik ve performansı sağlama olarak özetlenebilir.

Veritabanı yöneticisi ile veritabanı tasarımcısı arasındaki fark, rolleri ve sorumluluklarına dayanır. İşlevsel olarak:

1. \*\*Veritabanı Yöneticisi (Database Administrator - DBA):\*\*

- \*\*Erişim Kontrolü:\*\* DBA, veritabanına erişim yetkilerini tanımlar ve yönetir. Kullanıcıların hangi verilere erişebileceğini ve hangi işlemleri yapabileceğini belirler.

- \*\*Performans ve Güvenlik:\*\* Veritabanının güvenliğini sağlar ve performansını optimize eder. Günlük yedeklemeleri gerçekleştirir ve veritabanı hatalarını çözer.

- \*\*Yazılım ve Donanım Yönetimi:\*\* İhtiyaç duyulan yazılım ve donanım kaynaklarını yönetir ve temin eder. Veritabanı sistemlerinin kurulumunu, yapılandırmasını ve güncellemesini gerçekleştirir.

2. \*\*Veritabanı Tasarımcısı (Database Designer):\*\*

- \*\*Veri Modelleme:\*\* Veritabanının mantıksal yapısını tanımlar. Veriler arasındaki ilişkileri, veri tiplerini ve bütünlük kısıtlamalarını belirler.

- \*\*Veritabanı Şeması Oluşturma:\*\* Veritabanının fiziksel yapısını tasarlar. Tabloların, indekslerin ve diğer nesnelerin nasıl oluşturulacağını belirler.

- \*\*Kullanıcı İhtiyaçlarını Anlama:\*\* Potansiyel kullanıcı gruplarıyla iletişim kurarak onların ihtiyaçlarını belirler. Tasarımı, kullanıcıların gereksinimlerini karşılayacak şekilde optimize eder.

Bu iki rol genellikle aynı kişi veya ekip tarafından yerine getirilir, ancak büyük ölçekli organizasyonlarda farklı kişiler arasında bölünebilir. Veritabanı yöneticisi, veritabanının işletilmesi ve günlük yönetimiyle ilgilenirken, veritabanı tasarımcısı, veritabanının yapısal tasarımı ve kullanıcı gereksinimlerinin karşılanmasıyla ilgilenir.

Üç Şema Mimarisi, veritabanı sistemlerinin karmaşık yapısını anlamak ve yönetmek için önemli bir kavramdır. Bu mimari, veritabanının farklı kullanıcı grupları ve uygulamalar arasında ayrı ayrı düşünülmesini ve yönetilmesini sağlar. İşlevsel olarak şu şekilde özetlenebilir:

1. \*\*İçsel (Fiziksel) Düzey (Internal Level):\*\* Bu düzey, veritabanının fiziksel depolama yapısını tanımlar. Verilerin fiziksel olarak nasıl depolandığı, veritabanına erişim yolları ve depolama yöntemleri bu düzeyde belirlenir. İçsel düzey, veritabanının fiziksel gerçekliğini temsil eder.

2. \*\*Kavramsal Düzey (Conceptual Level):\*\* Kavramsal düzey, tüm veritabanının yapısını tanımlar. Kullanıcılar için veritabanının bütünsel yapısını gösterir ve fiziksel depolama detaylarını gizler. Veri tipleri, ilişkiler, kısıtlamalar ve diğer veritabanı özellikleri bu düzeyde belirlenir.

3. \*\*Dışsal (Görünüm) Düzey (External Level):\*\* Bu düzey, farklı kullanıcı gruplarının veya uygulamaların ihtiyaçlarına göre özelleştirilmiş görünümleri tanımlar. Her bir dışsal düzey, belirli bir kullanıcı grubunun veya uygulamanın gereksinimlerini karşılamak için oluşturulur. Bu düzey, kullanıcıların veriye erişimini kolaylaştırır ve veritabanının karmaşıklığını gizler.

Bu üç düzey arasında eşleştirmeler yapılır ve veri bağımsızlığı sağlanır. Veri bağımsızlığı, bir düzeyde yapılan değişikliklerin diğer düzeyleri etkilemeden yapılabilmesini sağlar. Mantıksal veri bağımsızlığı, kavramsal düzeydeki değişikliklerin dışsal düzeyi etkilemeden yapılabilmesini sağlar. Fiziksel veri bağımsızlığı ise içsel düzeydeki değişikliklerin kavramsal düzeyi etkilemeden yapılabilmesini sağlar.

Bu mimari, veritabanı sistemlerinin karmaşıklığını azaltır, kullanıcıların ve uygulamaların veriye erişimini kolaylaştırır ve veritabanının yönetimini daha etkili hale getirir.

Veritabanı Yönetim Sistemlerinde (VTYS) kullanılan diller, veritabanının tasarımından uygulanmasına kadar bir dizi farklı kullanım alanında ortaya çıkar. İşlevsel olarak şu şekilde özetlenebilir:

1. \*\*Veri Tanımlama Dili (Data Definition Language - DDL):\*\* Veritabanı şemalarını tanımlamak ve oluşturmak için kullanılan bir dildir. Kavramsal ve fiziksel şemalar arasındaki ilişkiyi belirlemek için kullanılır. Veri tanımlama dili, veritabanı yöneticileri ve tasarımcıları tarafından kullanılır ve VTYS tarafından işlenerek VTYS kataloğunda depolanacak yapılar haline dönüştürülür.

2. \*\*Depolama Tanımlama Dili (Storage Definition Language - SDL):\*\* Bazı VTYS'lerde fiziksel şemanın belirlenmesi için kullanılan bir dildir. Depolama yapısı, erişim yolları ve fiziksel depolama ayrıntıları gibi konuları tanımlar.

3. \*\*Görünüm Tanımlama Dili (View Definition Language):\*\* Üç şema mimarisini tam anlamıyla destekleyen VTYS'lerde kullanılan bir dildir. Kullanıcı görünümlerini belirlemek ve bunların kavramsal şemadaki eşlemelerini belirlemek için kullanılır.

4. \*\*Sorgulama Dili (Query Language):\*\* Veritabanından bilgi almak ve veri üzerinde işlem yapmak için kullanılan bir dildir. Kullanıcıların ve uygulamaların veriye erişimini sağlar. Örneğin, ilişkisel VTYS'lerde SQL (Structured Query Language) sıkça kullanılan bir sorgulama dilidir.

5. \*\*Veri İşleme Dili (Data Manipulation Language - DML):\*\* Veritabanı üzerinde veri ekleme, silme, güncelleme gibi işlemleri gerçekleştirmek için kullanılan bir dildir. VTYS'nin bu işlemleri gerçekleştirmesi için kullanılır.

Günümüzde, ilişkisel VTYS'ler genellikle SQL'i birleştirilmiş bir dil olarak kullanır. SQL, DDL, SDL, ve DML işlevlerini içeren geniş kapsamlı bir dildir. Bu dil, veritabanı tasarımından veri işlemeye kadar birçok farklı görevde kullanılabilir.

Veritabanı yönetim sistemlerinin bileşen modülleri, kullanıcı arayüzünden depolama ve işleme alt sistemlerine kadar geniş bir yelpazede yer alır. VTYS bileşenleri, veritabanı yönetimi için gerekli tüm işlevleri gerçekleştirmek üzere tasarlanmıştır. Bu bileşenler arasında veri tanımlama dili derleyicisi, sorgu işleyici, depolanmış veri yöneticisi gibi önemli modüller bulunur.

Veritabanı sistemlerindeki bu diller ve bileşenler, veritabanının etkin yönetimi ve kullanımı için temel öneme sahiptir. Bu bileşenlerin ve dillerin doğru ve etkili bir şekilde kullanılması, veritabanı performansını artırabilir ve veri yönetimini kolaylaştırabilir.

Veritabanı yönetim sistemlerinin kullanılan dilleri ve veritabanı türleri hakkında oldukça detaylı bir açıklama yapmışsınız! Veritabanı türleri, kullanılan veri modeline göre farklılık gösterir ve her bir türün avantajları ve dezavantajları vardır.

Hiyerarşik veritabanı, ağ veritabanı, ilişkisel veritabanı ve nesneye yönelik veritabanı gibi farklı veritabanı türlerine kısa ve net bir şekilde değinmişsiniz. Her biri belirli bir problem alanına uygun olduğu için tercih edilir. Örneğin, ilişkisel veritabanı modelleri, günümüzde en yaygın olarak kullanılan veritabanı türüdür.

Ayrıca, veritabanı yönetim sistemleri yazılımları hakkında da bilgi vermişsiniz. MS SQL Server, Oracle, MySQL, Sybase, MS Access, PostgreSQL, ve IBM DB2 gibi birçok farklı yazılım mevcuttur. Bu yazılımların her biri farklı özelliklere ve kullanım alanlarına sahiptir.

Tabii, işte verdiğiniz bilgilerin kısa ve öz bir özeti:

1. **Veritabanı Türleri:**
   * Hiyerarşik veritabanı: Kök ve dallar biçiminde, ağaç yapısına benzeyen bir yapıda verilerin saklandığı eski bir veri modelidir. IMS gibi yazılımlar tarafından kullanılır.
   * Ağ veritabanı: Hiyerarşik yapıdaki sınırlamaları gidermek için geliştirilmiştir. Her düğüm birden fazla ebeveyn ve çocuk düğümüne bağlanabilir.
   * İlişkisel veritabanı: İlk olarak 1970'lerde geliştirilmiş ve günümüzde en yaygın kullanılan modeldir. Birden çok ilişki biçimi kullanılabilir.
   * Nesneye yönelik veritabanı: Multimedya verilerini ve kompleks veri türlerini içeren modern uygulamalarda kullanılır.
2. **Veritabanı Yönetim Sistemleri Yazılımları:**
   * MS SQL Server: Microsoft tarafından geliştirilmiş, ilişkisel veri modelini kullanan ve yalnızca Windows işletim sisteminde çalışan bir yazılımdır.
   * Oracle: Kurumsal amaçlar için tercih edilen, büyük miktarda veriyi güvenli bir şekilde saklamak için kullanılan bir yazılımdır.
   * MySQL: Açık kaynak kodlu bir veritabanı yönetim sistemi yazılımıdır. Özellikle web uygulamalarında yaygın olarak kullanılır.
   * Sybase: SAP'a bağlı bir yazılım şirketi tarafından geliştirilen ve orta ile büyük ölçekli uygulamalar için tercih edilen bir yazılımdır.
   * PostgreSQL: Açık kaynak kodlu ve geniş özelliklere sahip bir yazılımdır. UNIX ve Windows işletim sistemlerinde çalışabilir.
   * MS Access: Microsoft Office paketi içinde yer alan küçük ölçekli masaüstü veritabanı yönetim sistemi yazılımıdır.
   * DB2: IBM tarafından geliştirilen, küçük ve büyük ölçekli uygulamalar için kullanılan bir yazılımdır.

1. \*\*Veri ve Veritabanı Tanımı\*\*:

- Veri, ham gözlemler, işlenmemiş gerçekler ya da izlenimlerdir. Veritabanı ise, birbiriyle ilişkili olan ve amaca uygun olarak düzenlenmiş, mantıksal ve fiziksel olarak tanımlanmış veriler bütünüdür.

2. \*\*Geleneksel Dosya Sistemleri ve Veritabanı Yönetim Sistemleri\*\*:

- Geleneksel dosya sistemlerinde veriler ayrı ayrı dosyalarda saklanırken, veritabanı yönetim sistemleri veri tekrarını ve tutarsızlığını önler ve veri güvenliği sağlar.

3. \*\*Veritabanı Kullanıcılarının Sınıflandırılması\*\*:

- Veritabanı yöneticisi, veritabanına erişim yetkilerini belirler ve güvenliği sağlar.

- Veritabanı tasarımcısı, verilerin tanımlanması ve yapıların seçilmesinden sorumludur.

- Son kullanıcılar, veritabanına sorgulama yapar veya raporlar oluşturur.

4. \*\*Veritabanı Sistemlerinin Mimarisi\*\*:

- Veri modeli, bir veritabanının mantıksal yapısını tanımlar ve genellikle üç düzeyde tanımlanır: içsel (fiziksel), kavramsal ve dışsal (görünüm) düzeyler.

- Veritabanı tanımları veri tanımlama dili kullanılarak oluşturulur ve veri işleme dili ile veri sorgulama yapılır. SQL (Structured Query Language), tipik bir ilişkisel veritabanı dilidir.

5. \*\*Veritabanı Türleri ve Yaygın Yazılımlar\*\*:

- Veritabanları, hiyerarşik, ağ, ilişkisel ve nesneye yönelik olarak sınıflandırılabilir.

- Yaygın olarak kullanılan veritabanı yönetim sistemleri arasında MS SQL Server, Oracle, MySQL, PostgreSQL ve diğerleri bulunur.

BİR >>>>>>N

* OKUL>>>>SINIFLAR
* OKUL>>>>ÖĞRETMENLER
* OKUL>>>>ÖĞRENCİLER
* ÖĞRETMEN>>>>> ÖĞRENCİLER

N>>>>>>>N

* ÖĞRETMENLER>>>>>>SINIFLAR
* DOKTORLAR >>>>>>HASTALAR
* MAHALLE>>>>>>EVLER

BİR>>>>>>>BİR

* DOKTOR>>>>HASTA
* ARABA>>>>>SÜRÜCÜ
* İŞVEREN>>>>>İŞÇİ

N>>>>>>>>>BİR

* ÖĞRENCİ>>>NOT
* VATANDAŞ>>>>>ÜLKE
* HASTALAR>>>>DOKTOR
* SINIF>>>>>>ÖĞRETMEN