

Bölüm 1 Veritabanı Sistemleri

1.Aşağıdaki kelimeleri tanımlayın:

- Veri (data):** Ham ve işlenmemiş bilgiler, tek başına anlam ifade etmeyen, ancak işlenerek bilgiye dönüştürülebilen öğelerdir.
- Alan (field):** Veritabanlarında, her kaydın bir özelliğini temsil eden küçük veri birimidir. Örneğin, bir müşteri kaydında müşteri adı, adresi veya telefon numarası bir alan olabilir.
- Kayıt (record):** Bir veritabanında, bir alanın bir araya gelerek oluşturduğu tek bir veri satırıdır. Örneğin, bir müşteri kaydında müşteri adı, adresi, telefon numarası bir araya gelir.
- Dosya (file):** Veritabanındaki kayıtları içeren ve genellikle belirli bir konuya ait olan veri kümesidir.

2.Veritabanı yedekleme (data redundancy) nedir ve dosya sisteminin hangi özellikleri buna yol açabilir?

Veritabanı yedekleme, verilerin birden fazla yerde tekrarlanması durumudur. Dosya sistemlerinde veri yedekleme, verilerin bağımsız dosyalarda tutulması, veri düzenlemelerinin zayıf olması ve verilerin güncellenmesi sırasında tutarsızlıkların ortaya çıkması gibi nedenlerle oluşabilir.

3.Veritabanı bağımsızlığı (data independence) nedir ve neden dosya sistemlerinde eksiktir?

Veritabanı bağımsızlığı, verilerin yapısındaki değişikliklerin uygulama programlarına yansımaması durumudur. Dosya sistemlerinde veri bağımsızlığı eksiktir çünkü veriler genellikle uygulamalarla sıkı bir şekilde bağlantılıdır ve yapıdaki değişiklikler, uygulamaları etkileyebilir.

4.Bir Veritabanı Yönetim Sistemi (VTYS) nedir ve işlevleri nelerdir?

VTYS, verileri depolamak, düzenlemek ve yönetmek için kullanılan yazılım sistemidir. İşlevleri arasında veritabanı tasarımı, veri güvenliği, veri bütünlüğü sağlama, veri yedekleme ve sorgulama işlemleri bulunur.

5.Yapısal bağımsızlık nedir ve neden önemlidir?

Yapısal bağımsızlık, veritabanındaki veri yapısındaki değişikliklerin, uygulama yazılımlarını etkilememesidir. Bu özellik, veritabanı yönetimini kolaylaştırır ve veri yapısındaki değişikliklere karşı esneklik sağlar.

6.Veritabanı, bilgi ve veritabanı arasındaki farkları açıklayın.

- Veri**, ham, işlenmemiş bilgi öğeleridir.
- Bilgi**, verilerin işlenip anlamlı hale getirilmiş halidir.
- Veritabanı**, verilerin organize bir şekilde depolandığı ve yönetildiği sistemdir.

7.Bir Veritabanı Yönetim Sistemi'nin rolü nedir ve avantajları nelerdir? Dezavantajları nelerdir?

Veritabanı Yönetim Sistemi, verileri yönetmek için kullanılır ve avantajları arasında veri güvenliği, veri bütünlüğü, esneklik ve merkezi yönetim bulunur. Dezavantajları arasında karmaşıklık ve yüksek maliyet sayılabilir.

8.Farklı veritabanı türlerini listeleyin ve açıklayın.

- İlişkisel veritabanı:** Verilerin tablolarda saklandığı ve SQL kullanılarak erişilen veritabanı türüdür.
- NoSQL veritabanı:** Yapısal olmayan verilerin depolandığı, genellikle büyük veri uygulamalarında kullanılan veritabanı türüdür.
- Dağıtık veritabanı:** Verilerin birden fazla yerde depolandığı ve yönetildiği veritabanı türüdür.

9. **Bir veritabanı sisteminin ana bileşenleri nelerdir?**

Bir veritabanı sistemi; veritabanı, VTYS, veritabanı uygulamaları ve veri yönetim araçlarından oluşur.

10. **Metadata nedir?**

Metadata, verinin içeriği, yapısı ve özellikleri hakkında bilgi sağlayan veridir.

11. **Veritabanı tasarımı neden önemlidir?**

Veritabanı tasarımı, verilerin verimli, güvenli ve doğru bir şekilde saklanmasını sağlar. İyi bir tasarım, veri yönetimini kolaylaştırır ve veri hatalarını önler.

12. **Bir veritabanı sistemi uygulamanın potansiyel maliyetleri nelerdir?**

Veritabanı uygulamanın maliyetleri arasında yazılım lisansları, donanım gereksinimleri, eğitim ve bakım maliyetleri sayılabilir.

13. **Yapılandırılmamış ve yapılandırılmış veriyi örneklerle karşılaştırın. Tipik bir iş ortamında hangi tür daha yaygındır?**

Yapılandırılmış veri, belirli bir formatta düzenlenmiş ve sorgulanabilir veridir (örneğin, tablolarda saklanan veriler). Yapılandırılmamış veri ise, düzenli bir yapıya sahip olmayan ve genellikle metin, görüntü veya ses gibi öğeler içeren veridir. Yapılandırılmış veri, tipik iş ortamlarında daha yaygındır.

14. **Bir elektronik tablonun gerçekleştiremeyeceği bazı temel veritabanı işlevleri nelerdir?**

Bir elektronik tablo, büyük veri kümelerinin yönetimi, veri bütünlüğü sağlama, karmaşık sorgular ve veritabanı güvenliği gibi temel veritabanı işlevlerini yerine getiremez.

15. **Son kullanıcılar tarafından oluşturulan elektronik tabloların, tipik bir dosya sistemiyle paylaştığı yaygın sorunlar nelerdir?**

Elektronik tablolarda veri tutarsızlıkları, veri çoğaltması, veri kaybı ve güvenlik sorunları yaygın olabilir.

16. **Bilgisayarlı veri depolarının ortaya çıkmasıyla birlikte, son kullanıcıların iş verilerine doğrudan erişimini kaybetmesinin önemini açıklayın.**

Veritabanı sistemleri, verilerin merkezi olarak yönetilmesini sağlar ve kullanıcıların doğrudan erişim yerine kontrollü erişim ile veri güvenliğini artırır.

17. **Bir bulut veritabanının sahip olma maliyetinin, geleneksel bir şirket veritabanına göre neden daha düşük olabileceğini açıklayın.**

Bulut veritabanları, donanım ve yazılım yatırımlarını ortadan kaldırarak, bakım maliyetlerini azaltır ve esnek ödeme planları sunar, bu da sahip olma maliyetlerini düşürür.

Bölüm 2 Veri Modelleri

1. **Veri modellerinin önemi nedir?**

Veri modelleri, veritabanlarının nasıl organize edileceğini ve veriler arasındaki ilişkilerin nasıl yapılandırılacağını belirler. İyi tasarlanmış veri modelleri, veritabanlarının verimli ve güvenli bir şekilde çalışmasını sağlar, veri yönetimini kolaylaştırır ve son kullanıcıların doğru verilere hızlı erişmesini mümkün kılar.

2. **İş kuralı nedir ve veri modellemedeki amacı nedir?**

İş kuralları, bir organizasyonun iş süreçlerini ve operasyonlarını yönlendiren yönergelerdir.

Veri modellemesinde, iş kuralları, verilerin nasıl düzenleneceğini, işlenmesi gerektiğini ve ilişkilerin nasıl kurulacağını belirler. Bu, veritabanı tasarımının doğru ve organizasyonel gereksinimlere uygun olmasına yardımcı olur.

3. **İş kurallarını veri modeli bileşenlerine nasıl dönüştürürsünüz?**

İş kuralları, veri modelindeki varlıklar (entiteler), ilişkiler (relationships) ve nitelikler (attributes) olarak dönüştürülür. İş kuralları, veri modelinin doğruluğunu sağlayan kısıtlamalar ve gereksinimler şeklinde ifade edilir.

4. **İlişkisel veri modelinin temel özelliklerini tanımlayabilecek ve bunların son kullanıcı ve tasarımcı için önemini tartışabilecektir.**

İlişkisel veri modelinin temel özellikleri arasında, verilerin tablolarda (ilişkilerde) depolanması, satırlar (kayıtlar) ve sütunlar (alanlar) şeklinde düzenlenmesi, verilerin anahtarlar kullanılarak birbirine bağlanması bulunur. Bu model, veri bütünlüğünü sağlamak ve verileri kolayca sorgulamak için önemli avantajlar sunar. Tasarımcılar için, ilişkisel modelin yapılandırılması daha düzenli veri depolama ve kolay veri erişimi sağlar. Son kullanıcılar ise SQL sorguları aracılığıyla veriye kolay erişebilir.

5. **Varlık ilişkileri (ER) modelinin daha yapılandırılmış bir ilişkisel veritabanı tasarım ortamının oluşturulmasına nasıl yardımcı olduğunu açıklayınız.**

Varlık ilişkileri (ER) modeli, veritabanındaki varlıklar ve bu varlıklar arasındaki ilişkileri tanımlar. ER modelini kullanarak, tasarımcılar veri yapısının daha düzenli ve mantıklı bir şekilde yapılandırılmasını sağlarlar. ER modeli, daha sonra ilişkisel veritabanı yapısına dönüştürülebilir.

6. **"Bir müşteri çok sayıda ödeme yapabilir, ancak her ödeme sadece bir müşteri tarafından yapılır." Bu senaryoyu bir varlık ilişki diyagramı (ERD) gösterimi için temel olarak kullanın.**

Bu senaryoda, **Müşteri** ve **Ödeme** varlıkları vardır. **Müşteri** varlığı ile **Ödeme** varlığı arasında bir **1:N** ilişkisi vardır (bir müşteri birden çok ödeme yapabilir, ancak her ödeme yalnızca bir müşteriyle ilişkilidir).

7. **Neden bir nesnenin bir varlıktan daha fazla anlamsal içeriğe sahip olduğu söylenir?**

Bir nesne, sadece veri içermekle kalmaz, aynı zamanda o veriyi işleyebilecek işlevleri ve metotları da içerir. Yani, bir nesne, bir varlıkla kıyaslandığında daha fazla anlam taşır çünkü sadece veri saklamakla kalmaz, aynı zamanda bu verilerle etkileşimde bulunabilir. Varlık, yalnızca bir veri ögesini temsil ederken, nesne, verinin işlenmesini ve yönetilmesini sağlar.

8. **Nesne yönelimli veri modelinde (OODM) bir nesne ile bir sınıf arasındaki fark nedir?**

Bir **sınıf** (class), benzer nesnelerin özelliklerini tanımlayan bir şablondur. Bir **nesne** (object), sınıfın örneğidir ve o sınıfın belirli bir durumu ile ilişkilidir. Sınıf, nesnelerin yapısını tanımlarken, nesne o yapıya göre oluşturulmuş ve işlevsel hale gelmiş bir varlıktır.

9. **Soru 6'yı bir OODM ile nasıl modellersiniz? (Kılavuz olarak Şekil 2.4'ü kullanın.)**

Bir OODM'de, **Müşteri** ve **Ödeme** arasındaki ilişkiyi, müşteri sınıfını ve ödeme sınıfını tanımlayarak modelleyebilirsiniz. **Müşteri** sınıfı, her müşteriye ait özellikleri (isim, adres, vs.) içerirken, **Ödeme** sınıfı, ödeme tarihi, tutar gibi özellikleri içerir. Bu iki sınıf arasında bir ilişki (association) kurarak, bir müşterinin birden fazla ödemesi olabileceğini belirtebiliriz.

10. **ERDM nedir ve modern (üretim) veritabanı ortamında nasıl bir rol oynar?**

ERDM, **Varlık İlişkisi Diyagramı Modeli** (Entity-Relationship Data Model) anlamına gelir. Modern veritabanı ortamlarında, ERDM, veritabanı tasarımını görselleştirerek, veritabanı

bileşenlerinin birbirleriyle nasıl ilişkili olduğunu açıkça gösterir. Bu, tasarımcıların veritabanlarını daha doğru ve verimli bir şekilde tasarlamalarına yardımcı olur.

11. İlişki nedir ve hangi üç tür vardır?

İlişki, veritabanında iki varlık arasındaki bağlantıyı temsil eder. Üç temel ilişki türü şunlardır:

- **1:1 (Bir-bir) ilişkisi:** Bir varlık örneği, yalnızca bir başka varlık örneğiyle ilişkilidir.
- **1:N (Bir-çok) ilişkisi:** Bir varlık örneği, birden fazla başka varlık örneğiyle ilişkilidir.
- **M:N (Çok-çok) ilişkisi:** Bir varlık örneği, birden fazla başka varlık örneğiyle ilişkilidir ve aynı şekilde, diğer varlık örnekleri de çok sayıda ilişki kurar.

12. Üç ilişki türünün her birine bir örnek veriniz.

- **1:1 (Bir-bir) ilişkisi:** Bir **kişinin** yalnızca bir **pasaportu** vardır ve her pasaport yalnızca bir kişiye aittir.
- **1:N (Bir-çok) ilişkisi:** Bir **öğretmen**, birden fazla **ders** verebilir, ancak her ders yalnızca bir öğretmen tarafından verilmiştir.
- **M:N (Çok-çok) ilişkisi:** Bir **öğrenci** birçok **ders** alabilir ve her ders, birçok **öğrenci** tarafından alınabilir.

13. Tablo nedir ve ilişkisel modelde nasıl bir rol oynar?

Tablo, ilişkisel veritabanında verilerin saklandığı bir yapıdır. Her tablo, belirli bir veri türünü temsil eden satırlar (kayıtlar) ve sütunlardan (alanlardan) oluşur. İlişkisel modelde tablo, verileri düzenlemek ve ilişkileri tanımlamak için kullanılır. Veritabanı tasarımında tablolar, verileri düzenli ve anlamlı bir şekilde saklamak için önemlidir.

14. İlişkisel diyagram nedir? Bir örnek veriniz.

İlişkisel diyagram (ERD), veritabanı bileşenleri ve bu bileşenler arasındaki ilişkileri görsel olarak temsil eden bir diyagramdır. Örnek: **Müşteri** ve **Sipariş** tabloları arasında bir ilişkiyi gösteren bir diyagram, **Müşteri** tablosunun **Sipariş** tablosuyla olan bir **1:N** ilişkisini gösterebilir.

15. Bağlanabilirlik nedir? (Bağlanabilirliği göstermek için bir Karga Ayağı ERD'si kullanın).

Bağlanabilirlik, veritabanında bir varlıkla diğer varlıklar arasındaki ilişkiyi tanımlar. "Karga Ayağı ERD'si", bir ilişkiyi tanımlarken, her varlığın diğerleriyle olan olası ilişkilerini açıklamak için kullanılan bir işarettir. Bu tür bir ERD'de, her varlık (örneğin, **Müşteri** ve **Sipariş**) aralarındaki ilişkiyi göstermek için çizgilerle bağlanır.

16. Büyük Veri olgusunu tanımlayın.

Büyük Veri (Big Data), geleneksel veri işleme araçlarıyla yönetilemeyecek kadar büyük, karmaşık ve hızlı veri kümelerini ifade eder. Bu veriler genellikle çok yüksek hacimli, hızla artan ve farklı türde veri içerir (yapısal, yarı yapısal ve yapısız).

17. 3 Vs terimi neyi ifade eder?

3 Vs, Büyük Veri'nin temel özelliklerini ifade eder:

- **Volume (Hacim):** Verinin büyüklüğü ve miktarı.

- **Velocity (Hız):** Verinin hızla üretilmesi ve işlenmesi.
- **Variety (Çeşitlilik):** Verinin farklı biçimlerde olması (yapısal, yarı yapısal, yapısız).

18. Hadoop nedir ve temel bileşenleri nelerdir?

Hadoop, büyük veri kümelerini depolamak ve işlemek için açık kaynaklı bir yazılım çerçevesidir. Temel bileşenleri:

- **Hadoop Distributed File System (HDFS):** Verilerin dağıtık bir şekilde depolandığı dosya sistemi.
- **MapReduce:** Verilerin paralel olarak işlenmesini sağlayan bir işleme modelidir.
- **YARN (Yet Another Resource Negotiator):** Kaynak yönetimi ve görev takibini sağlar.
- **Hadoop Common:** Hadoop'un diğer bileşenleriyle çalışması için gerekli temel araçlar ve kütüphaneler.

19. NoSQL veritabanının temel özellikleri nelerdir?

NoSQL veritabanları, büyük veri kümeleri için esnek, ölçeklenebilir ve hızlı veri yönetimi sağlayan sistemlerdir. Temel özellikleri:

- **Esneklik:** Veri modelleri için şemalar gerektirmez.
- **Yüksek ölçeklenebilirlik:** Veri kümesi büyüdükçe sistemin performansını artırabilir.
- **Dağıtık yapı:** Veriler farklı sunucularda paralel olarak depolanabilir.
- **Farklı veri yapıları:** Anahtar-değer, belge, sütunlu ve grafik tabanlı veri yapıları destekler.

20. Hastaları ve testleri olan bir tıp kliniği örneğini kullanarak, ilişkisel modeli kullanarak bu örneğin nasıl modelleneyeceğine dair basit bir gösterim sununuz.

Bir tıp kliniği örneğinde, şu şekilde ilişkiler kurulabilir:

- **Hasta** (Hasta_ID, Ad, Soyad, Doğum_Tarihi)
- **Test** (Test_ID, Test_Adı, Tarih)
- **Test_Sonuçları** (Sonuç_ID, Test_ID, Hasta_ID, Sonuç) Bu yapıda, bir hasta birden fazla teste sahip olabilir ve her test bir hasta ile ilişkilidir.

21. Mantıksal bağımsızlık nedir?

Mantıksal bağımsızlık, veritabanı tasarımının, kullanıcı uygulamalarından bağımsız olmasını ifade eder. Bu, veritabanı yapısındaki değişikliklerin, kullanıcıların uygulamalarını etkilememesini sağlar.

22. Fiziksel bağımsızlık nedir?

Fiziksel bağımsızlık, verilerin fiziksel depolama biçimindeki değişikliklerin, veritabanı uygulamalarını etkilememesini ifade eder. Bu, verilerin depolanma şeklinde yapılan değişikliklerin, kullanıcıların veri erişimini etkilememesini sağlar.