

Veri Tabanı Temelleri

1. NoSQL veri tabanı nedir? Özellikleri nelerdir?

NoSQL (Not Only SQL), ilişkisel olmayan veri tabanlarını tanımlayan bir terimdir. Bu veri tabanları, yapılandırılmamış veya yarı yapılandırılmış verileri depolamak için esnek veri modelleri sunar. Belgeler (Document), anahtar-değer çiftleri (Key-Value), sütun tabanlı (Column-Based) ve grafik tabanlı (Graph) olmak üzere farklı türleri vardır. NoSQL sistemler, yatayda ölçeklenebilirlik, yüksek performans ve büyük veri (big data) desteği gibi özellikler sunar.

2. SQL Server Veri Tipleri nelerdir? (Örneklerle açıklayınız.)

SQL Server'da verilerin türüne göre birçok farklı veri tipi kullanılır. Bu veri tipleri aşağıdaki ana başlıklar altında gruplanabilir:

a) Sayısal Veri Tipleri:

- INT: Tamsayı değerler için. Örnek: INT → 100
- SMALLINT: Daha küçük tamsayılar için. Örnek: SMALLINT → 32000
- BIGINT: Çok büyük tamsayılar için. Örnek: BIGINT → 9223372036854775807
- TINYINT: 0–255 arası sayılar için. Örnek: TINYINT → 200
- DECIMAL(p,s) veya NUMERIC(p,s): Kesirli ondalıklı sayılar için. Örnek: DECIMAL(10,2) → 1234.56
- FLOAT: Kayan noktalı sayılar için. Örnek: FLOAT → 3.1415926535
- REAL: Daha düşük hassasiyetli kayan nokta. Örnek: REAL → 1.23

b) Metin (Karakter) Veri Tipleri:

- CHAR(n): Sabit uzunlukta metin. Örnek: CHAR(10) → 'Merhaba '
- VARCHAR(n): Değişken uzunlukta metin. Örnek: VARCHAR(50) → 'OpenAI'
- TEXT: Büyük miktarda metin. (Kullanımı eski, yerine VARCHAR(MAX) önerilir)

c) Unicode Karakter Tipleri (Çok dilli destek için):

- NCHAR(n): Sabit uzunlukta Unicode karakterler.

- NVARCHAR(n): Değişken uzunlukta Unicode karakterler.
- NTEXT: Büyük miktarda Unicode metin (kullanımı önerilmez, NVARCHAR(MAX) tercih edilir)

d) Tarih ve Zaman Veri Tipleri:

- DATE: Sadece tarih. Örnek: DATE → '2025-05-08'
- TIME: Sadece saat. Örnek: TIME → '14:30:00'
- DATETIME: Tarih ve saat. Örnek: DATETIME → '2025-05-08 14:30:00'
- DATETIME2: DATETIME'e göre daha hassas ve geniş kapsamlı.
- SMALLDATETIME: Daha az hassas tarih-saat tipi.
- DATETIMEOFFSET: Saat dilimi bilgisiyle birlikte tarih-saat.

e) Mantıksal Veri Tipi:

- BIT: 0 veya 1 değerini alabilir. Örnek: BIT → 1 (true)

3. CRUD İşlemleri nedir? (Create, Read, Update, Delete)

CRUD, veri tabanındaki temel işlemleri tanımlayan dört ana işlemidir:

- **Create:** Yeni veri eklemek (INSERT)
- **Read:** Var olan veriyi okumak (SELECT)
- **Update:** Mevcut veriyi güncellemek (UPDATE)
- **Delete:** Veriyi silmek (DELETE)

Bu işlemler, verilerin yönetimi için temel yapıyı oluşturur ve her veri tabanı uygulamasında kullanılır.

4. Foreign Key nedir? Nasıl kullanılır?

Foreign Key (Yabancı Anahtar), bir tablodaki sütunun başka bir tablodaki birincil anahtara (Primary Key) referans vermesini sağlar. Bu, tablolar arasında ilişki kurmak ve veri bütünlüğünü sağlamak için kullanılır.

Örnek: Orders tablosundaki CustomerID alanı, Customers tablosundaki ID alanına bağlı bir foreign key olabilir. Bu ilişki sayesinde, silinmemesi gereken ya da eşleşmeyen verilerin önüne geçilir.

5. Join işlemleri nedir? Çeşitleri nelerdir? (Inner Join, Left Join, vb.)

Join işlemleri, birden fazla tabloyu ortak alanlar üzerinden birleştirmek için kullanılır. Başlıca Join türleri:

- **INNER JOIN:** Her iki tabloda eşleşen kayıtları döndürür.
- **LEFT JOIN (LEFT OUTER JOIN):** Sol tablodaki tüm verileri ve sağ tabloda eşleşen verileri getirir.
- **RIGHT JOIN (RIGHT OUTER JOIN):** Sağ tablodaki tüm verileri ve sol tabloda eşleşen verileri getirir.
- **FULL JOIN (FULL OUTER JOIN):** Her iki tablodaki tüm verileri, eşleşsin ya da eşleşmesin döndürür.
Join işlemleri, ilişkili veri kümeleriyle çalışmayı kolaylaştırır ve raporlama için sıkça kullanılır.