

데이터베이스 핵심개념 정리

1. 데이터베이스 (Database)



(이미지출처) <https://www.datamation.com/big-data/top-database-challenges/>

데이터베이스는 관련 데이터를 체계적으로 수집, 저장, 관리, 검색, 수정할 수 있는 시스템입니다. 데이터베이스 관리 시스템(DBMS)을 통해 이러한 데이터베이스를 효율적으로 관리하고 운영할 수 있습니다. 데이터베이스는 IT, 비즈니스, 의료, 교육, 정부 등 다양한 분야에서 중요한 역할을 하며 널리 사용되고 있습니다.

- 1) **구조화된 데이터**: 데이터는 테이블, 행, 열의 형태로 구조화되어 있어서 검색과 관리가 용이합니다.
- 2) **데이터 무결성과 일관성**: 데이터의 정확성과 일관성을 유지하기 위한 규칙과 제약 조건을 적용할 수 있습니다.
- 3) **효율적인 데이터 관리**: 대량의 데이터를 효율적으로 저장하고 관리할 수 있습니다.
- 4) **동시성 제어**: 여러 사용자가 동시에 데이터베이스에 접근하더라도 안정적인 데이터 처리가 가능합니다.

5) 보안: 권한 관리를 통해 데이터에 대한 접근을 제어할 수 있어 정보의 보안을 유지할 수 있습니다.

6) 데이터 접근성: 언제 어디서나 필요한 데이터에 접근할 수 있습니다.

7) 데이터 분석 및 의사결정 지원: 데이터를 분석하여 비즈니스 의사결정에 필요한 인사이트를 제공합니다.

8) 업무 자동화 및 효율성 증대: 반복적인 업무를 자동화하여 업무 효율성을 향상시킬 수 있습니다.

9) 데이터 보호 및 백업: 중요한 데이터를 안전하게 보호하고 재해 발생 시 복구할 수 있습니다.

10) 정보 통합: 다양한 정보원에서 수집한 데이터를 하나의 데이터베이스에 통합하여 관리할 수 있습니다.

2. 관계형 데이터베이스 , 비 관계형 데이터베이스

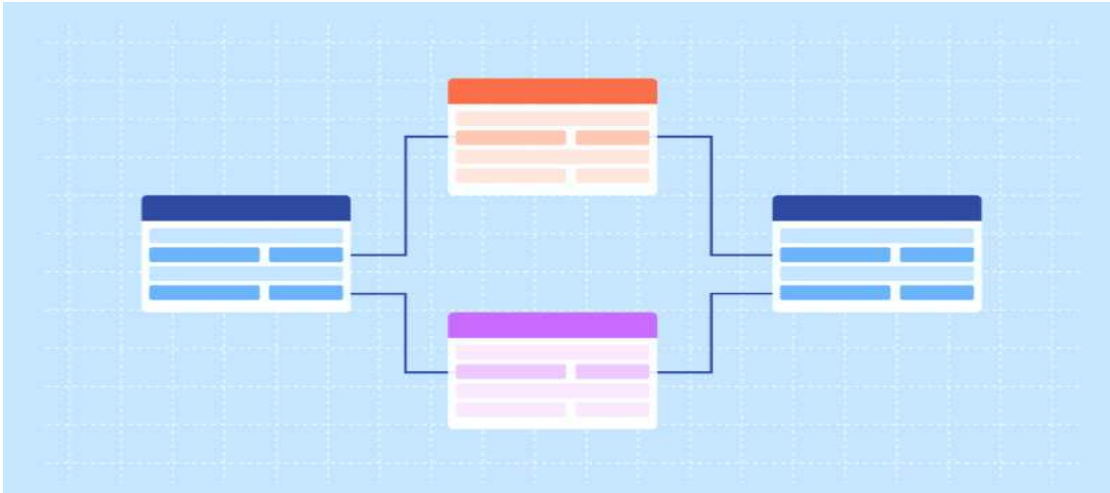
관계형 데이터베이스는 데이터를 테이블과 열로 구조화하고 엄격한 스키마를 가지며, 비 관계형 데이터베이스는 다양한 데이터 형태를 유연하게 다루며 스키마를 동적으로 조정할 수 있습니다. 데이터의 종류와 요구 사항에 따라 적절한 데이터베이스를 선택할 수 있습니다.

(관계형 데이터베이스 , 비관계형 데이터베이스)

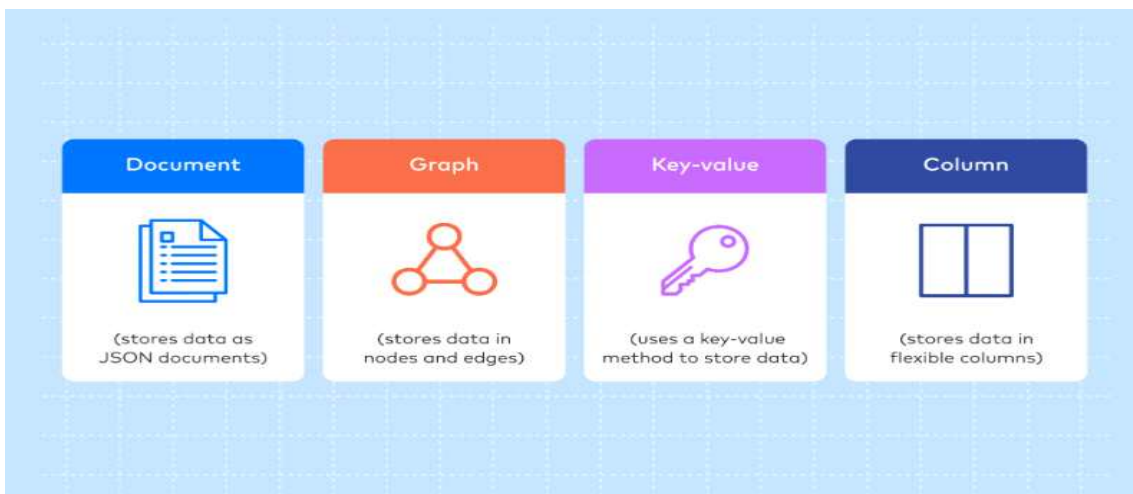


(이미지 출처) <https://www.fivetran.com/blog/what-is-a-database>

(관계형 데이터베이스)



(비 관계형 데이터 베이스)



1) 데이터 모델

관계형 데이터베이스: 테이블과 열을 사용하여 데이터를 저장합니다. 이것은 마치 엑셀 스프레드시트와 비슷한 형태입니다.

비 관계형 데이터베이스: 다양한 형태의 데이터 모델을 사용합니다. 데이터를 테이블로 구조화하지 않고도 저장할 수 있습니다.

2) 스키마 설계

관계형 데이터베이스: 정적인 스키마를 가지며, 데이터 구조가 미리 정의되어야 합니다. 필요한 경우 스키마를 변경하기 어려울 수 있습니다.

비 관계형 데이터베이스: 동적인 스키마를 사용하며, 필요에 따라 데이터 모델을 유연하게 수정할 수 있습니다.

3) 쿼리 언어

관계형 데이터베이스: SQL (Structured Query Language)을 사용하여 데이터를 검색하고 조작합니다.

비 관계형 데이터베이스: 다양한 NoSQL 데이터베이스는 SQL과 다른 쿼리 언어를 사용하며, 데이터를 다루기 위한 고유한 방식을 제공합니다.

4) 확장성

관계형 데이터베이스: 수직 확장은 가능하지만, 수평 확장이 어려우며 비용이 많이 들 수 있습니다.

비 관계형 데이터베이스: 수평 확장이 쉽고 비용 효율적이며, 대규모 데이터와 트래픽 처리에 더 적합합니다.

5) 일관성

관계형 데이터베이스: ACID 트랜잭션을 사용하여 데이터 일관성을 엄격하게 보장합니다.

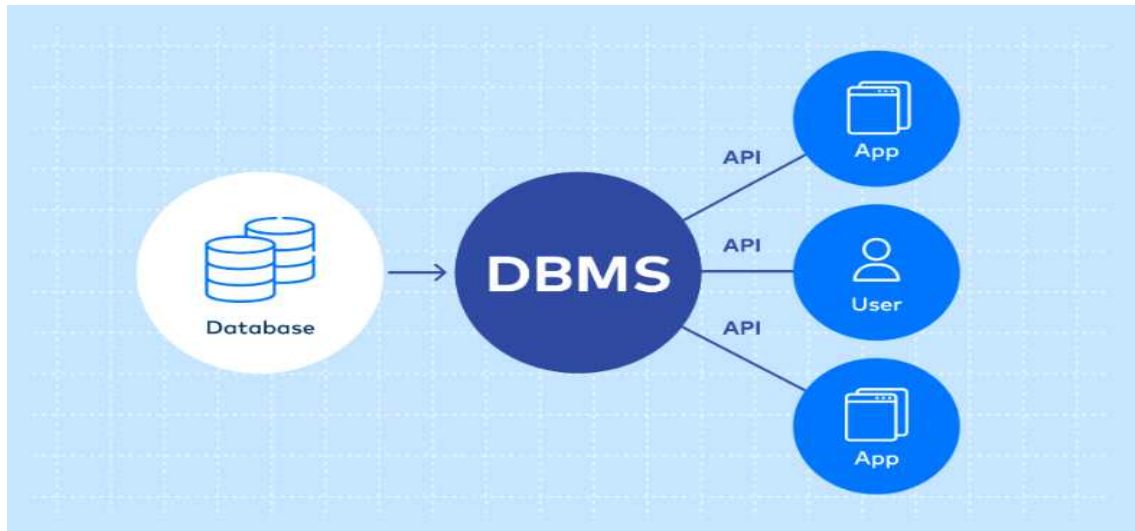
비 관계형 데이터베이스: NoSQL 데이터베이스는 데이터 일관성을 더 유연하게 다룰 수 있으며, 일부는 ACID를 제한적으로 지원합니다.

6) 속도

관계형 데이터베이스: 복잡한 조인 및 트랜잭션 처리로 인해 대용량 데이터에 대한 응답 시간이 상대적으로 느릴 수 있습니다.

비 관계형 데이터베이스: 간단한 읽기 및 쓰기 작업을 빠르게 처리하며, 대용량 데이터 처리에 더 적합합니다.

3. DBMS(DataBase Management System)



DBMS는 "데이터베이스 관리 시스템"의 줄임말로, 데이터를 효과적으로 저장, 관리 및 검색할 수 있도록 도와주는 소프트웨어나 시스템입니다. DBMS는 데이터를 효율적으로 관리하고 보호하기 위한 도구로서, 비즈니스나 개인이 데이터를 안전하게 저장하고 활용할 수 있도록 도와주는 중요한 시스템입니다.

DBMS는 데이터를 마치 컴퓨터의 파일처럼 테이블 형태로 정리하여 저장합니다. 각 테이블은 데이터의 특정 부분을 나타내며, 이러한 테이블들은 관계형 데이터베이스라고 불립니다. 각 테이블에는 행과 열이 있는데, 행은 각각의 데이터 레코드를 나타내고, 열은 해당 레코드의 속성이나 정보를 나타냅니다.

- 1) **데이터 저장:** 데이터를 안전하게 저장하고 필요할 때 빠르게 접근할 수 있도록 합니다.
- 2) **데이터 검색 및 조회:** 저장된 데이터를 검색하고 원하는 정보를 찾아볼 수 있게 도와줍니다. SQL(Structured Query Language)이라는 언어를 사용하여 데이터베이스에서 검색 및 조작 작업을 수행합니다.
- 3) **데이터 무결성 유지:** 데이터베이스 내의 데이터가 일관성 있게 유지되도록 보장하며, 잘못된 데이터가 저장되는 것을 방지합니다.
- 4) **보안:** 데이터베이스 접근 권한을 관리하여 민감한 정보를 보호합니다.
- 5) **동시성 제어:** 여러 사용자가 동시에 데이터베이스에 접근할 때 데이터 충돌을 방지하고 일관성을 유지합니다.

6) 데이터 백업 및 복구: 데이터 손실을 방지하기 위해 주기적으로 백업을 생성하고, 문제 발생 시 데이터를 복구할 수 있도록 합니다.

7) 성능 최적화: 데이터베이스 쿼리 및 작업을 최적화하여 빠른 응답 시간을 제공합니다.

4. ACID

ACID는 데이터베이스 관리 시스템(DBMS)에서 중요한 데이터 일관성과 안정성을 보장하기 위한 속성의 약어입니다. ACID 속성을 준수하는 것은 데이터베이스 시스템에서 데이터의 무결성과 안정성을 보장하며, 중요한 비즈니스 트랜잭션을 처리하는 데 필수적입니다.

1) 원자성 (Atomicity)

이해: 트랜잭션(Transaction)은 하나의 작업 또는 여러 작업으로 구성되어 있을 수 있습니다. 원자성은 트랜잭션 내의 모든 작업이 성공하거나 실패할 때 전체 트랜잭션이 성공 또는 실패해야 함을 의미합니다.

예시: 은행 계좌 이체 작업에서 돈을 보내는 쪽에서 돈을 빼고, 받는 쪽에서 돈을 받을 때 어느 한 쪽이 실패하면 전체 거래가 실패합니다.

2) 일관성 (Consistency)

이해: 트랜잭션이 실행되기 전과 후에 데이터베이스는 항상 일관된 상태여야 합니다. 즉, 트랜잭션이 데이터베이스를 손상시키면 안 됩니다.

예시: 잔고가 음수로 떨어지거나, 제약 조건에 어긋나는 데이터베이스 변경은 트랜잭션 내에서 일어나서는 안 됩니다.

3) 고립성 (Isolation)

이해: 다수의 트랜잭션이 동시에 실행되더라도 각 트랜잭션은 다른 트랜잭션의 작업에 영향을 받지 않아야 합니다. 트랜잭션은 격리되어야 합니다.

예시: 한 사용자가 계좌 잔액을 조회하는 동안, 다른 사용자의 이체 트랜잭션은 이 조회 작업에 영향을 주면 안 됩니다.

4) 지속성 (Durability)

이해: 트랜잭션이 성공적으로 완료되면 그 결과는 영구적으로 데이터베이스에 저장되어야 합니다. 시스템 장애 또는 전원 차단과 같은 문제가 발생해도 데이터는 안전하게 보존되어야 합니다.

예시: 트랜잭션에서 돈을 이체한 경우, 이 정보는 데이터베이스에 저장되어야 하며, 시스템 장애로 인해 데이터 손실이 발생하지 않아야 합니다.

5. 다양한 DataBase 특징



1) MySQL

특징: 오픈 소스 관계형 데이터베이스 관리 시스템(RDBMS)으로, 무료로 사용 가능하며 가볍고 빠르게 동작합니다.

용도: 웹 애플리케이션, 데이터 분석, 포털 등 다양한 응용 프로그램에 적합합니다.

장점: 저렴하거나 무료로 사용 가능하며, 커뮤니티 지원이 활발하며, 초보자에게 쉽게 배울 수 있습니다.

2) Oracle

특징: 상업용 데이터베이스 시스템으로, 대규모 엔터프라이즈 레벨의 애플리케이션과 데이터 처리에 적합합니다.

용도: 대규모 엔터프라이즈 애플리케이션, 데이터 웨어하우스, 빅 데이터 처리 등에 사용됩니다.

장점: 고성능, 안정성, 보안 기능이 강화되어 있으며, 대용량 데이터 처리에 특화되어 있습니다.

4) PostgreSQL

특징: 오픈 소스 RDBMS로, 확장성과 풍부한 기능을 제공하며 ACID 호환성을 유지합니다.

용도: 대규모 애플리케이션, 데이터 웨어하우스, 지리 정보 시스템 등 다양한 분야에서 사용됩니다.

장점: 높은 확장성, 고급 기능 지원, 커스터마이징 가능하며, 커뮤니티 지원이 활발합니다.

5) SQLite

특징: 경량의 오픈 소스 데이터베이스 엔진으로, 서버 없이 사용 가능하며 파일 기반 데이터베이스입니다.

용도: 모바일 애플리케이션, 임베디드 시스템, 데스크톱 애플리케이션, 작은 규모 웹 앱에 적합합니다.

장점: 무겁지 않고 사용이 간편하며, 서버가 필요 없어 포팅이 쉽습니다.

6) Microsoft SQL Server (MSSQL)

특징: Microsoft의 상업용 RDBMS로, Windows 플랫폼에 최적화되어 있으며 통합 보안 및 비즈니스 인텔리전스 기능을 제공합니다.

용도: 비즈니스 애플리케이션, 엔터프라이즈 애플리케이션, 비즈니스 인텔리전스 및 데이터 웨어하우스 등에 사용됩니다.

장점: Windows 생태계와 통합이 용이하며, 보안 및 기업 수준의 지원을 제공합니다. 각 데이터베이스 시스템은 특정 요구 사항과 사용 사례에 따라 선택되며, 가격, 성능, 기능, 커뮤니티 지원 등을 고려하여 선택해야 합니다.