

DATABASE

CONTENTS



01

데이터베이스 개념

02

DBMS

03

RDBMS vs NoSQL

04

MongoDB

05

DB Computer Science

'데이터베이스' 란?

데이터베이스(Database, DB)는 여러 사람이 공유하고 사용할 목적으로 통합 관리되는 정보의 집합이다.

데이터베이스 특징

실시간 접근성 (real time accessibility)
계속적인 진화 (continuous evolution)
동시 공유 (concurrent sharing)
내용에 의한 참조 (content reference)
데이터 논리적 독립성 (independence)

데이터베이스 장점

독립성 (data independence)
무결성 (data integrity)
보안성 (data security)
일관성 (data consistency)
중복 최소화 (avoiding data redundancy)

'DBMS' 란?

데이터베이스 관리 시스템(DataBase Management System)은 데이터베이스를 조작하는 별도의 소프트웨어

DBMS 기능

정의 , 구축, 조작, 공유, 보호, 유지보수

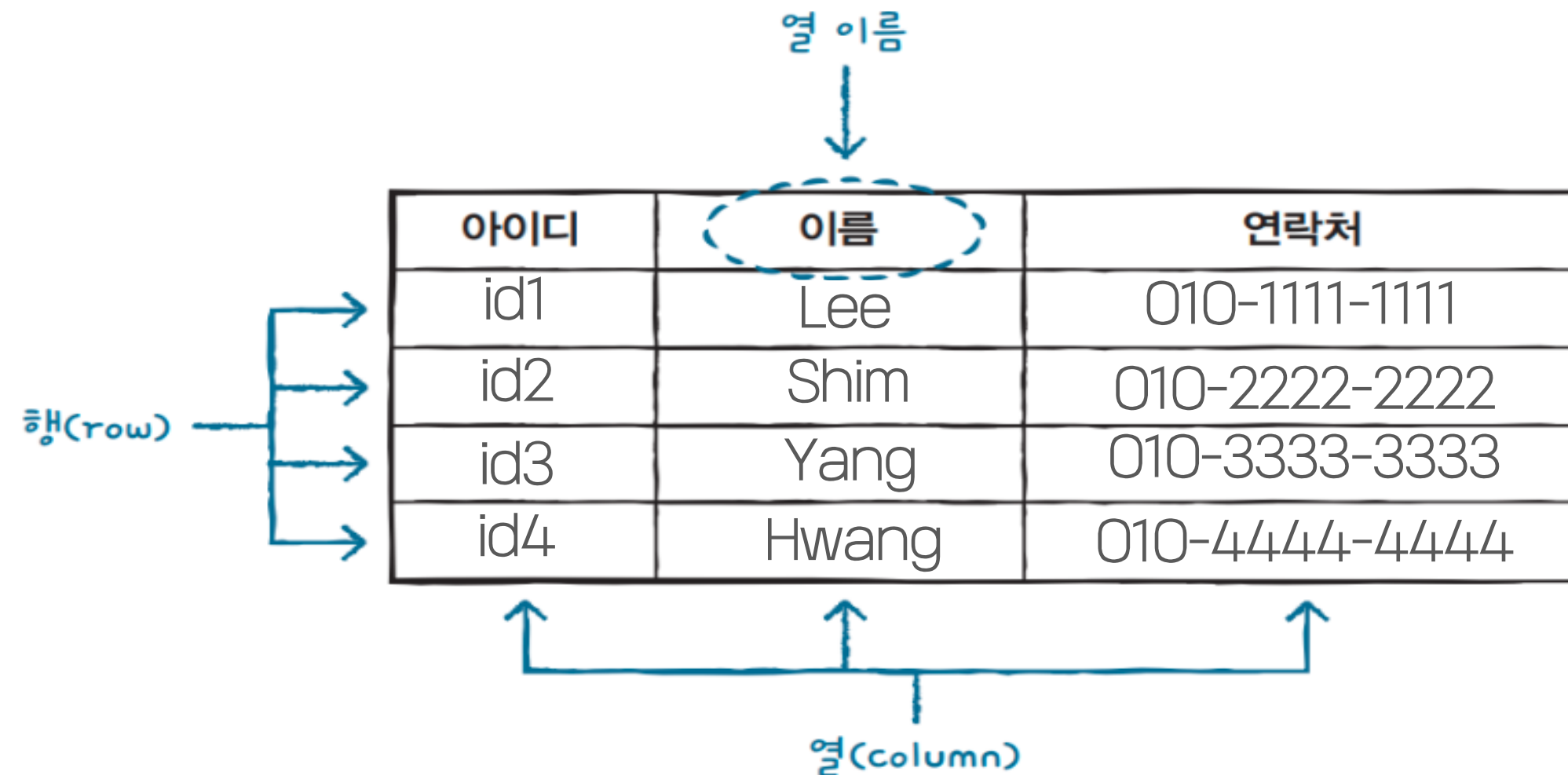
DBMS 종류

Oracle	MySQL	MSSQL	MariaDB
<ul style="list-style-type: none">- 오라클에서 만들어 판매중인 상업용 데이터베이스- 윈도우, 리눅스, 유닉스 등 다양한 운영체제(OS)에서 설치 가능- MySQL, MSSQL보다 대량의 데이터 처리 용이- 대기업에서 주로 사용하며, 글로벌 DB 시장 점유율 1위- 비공개 소스, 폐쇄적인 운영- 가장 널리 사용되는 RDBMS	<ul style="list-style-type: none">- MySQL사에서 개발, 썬마이크로시스템즈를 거쳐 현재 오라클에 인수합병- 윈도우, 리눅스, 유닉스 등 다양한 운영체제(OS)에서 설치 가능- 오픈소스로 이루어져있는 무료 프로그램(상업적 사용 시 비용 발생)- 가격 등의 장점을 앞세워 다수의 중소기업에서 사용중- RDBMS	<ul style="list-style-type: none">- 마이크로소프트(MS)사에서 개발한 상업용 데이터베이스- 다른 운영체제에서도 사용가능하지만 윈도우에 특화됨- 비공개 소스로 폐쇄적인 운영(리눅스 버전은 오픈소스)- 중소기업에서 주로 사용중- RDBMS	<ul style="list-style-type: none">- MySQL이 오라클에 인수합병된 후 불확실한 라이선스 문제를 해결하려고 나온 오픈소스 RDBMS- 구현언어 : C++- MySQL과 동일한 소스 코드 기반- MySQL과 비교해 애플리케이션 부분 속도가 약 4~5천 배 정도 빠름

DBMS 분류

계층형(Hierarchical), 망형(Network), 관계형(Relational), 객체지향형(Object-Oriented), 객체관계형(Object-Relational) 등으로 분류

관계형 DBMS



RDBMS vs NoSQL

	RDBMS	NoSQL
적합한 사용 예시	데이터 정합성이 보장되어야 하는 은행 시스템	낮은 지연 시간, 가용성이 중요한 SNS 시스템
데이터 모델	정규화와 참조 무결성이 보장된 스키마	스키마가 없는 자유로운 데이터 모델
트랜잭션	강력한 ACID 지원	완화된 ACID(BASE)
확장	하드웨어 강화(Scale up)	수평 확장 가능한 분산 아키텍처(Scale out)
API	SQL 쿼리	객체 기반 API 제공

Mongo DB

. NoSQL 데이터베이스이며 도큐먼트 지향 데이터베이스 시스템

Mongo DB 특징

Document-Oriented Storage

Full Index Support

Replication & High Availability

Auto-Sharding

Querying

Fast In-Place Updates

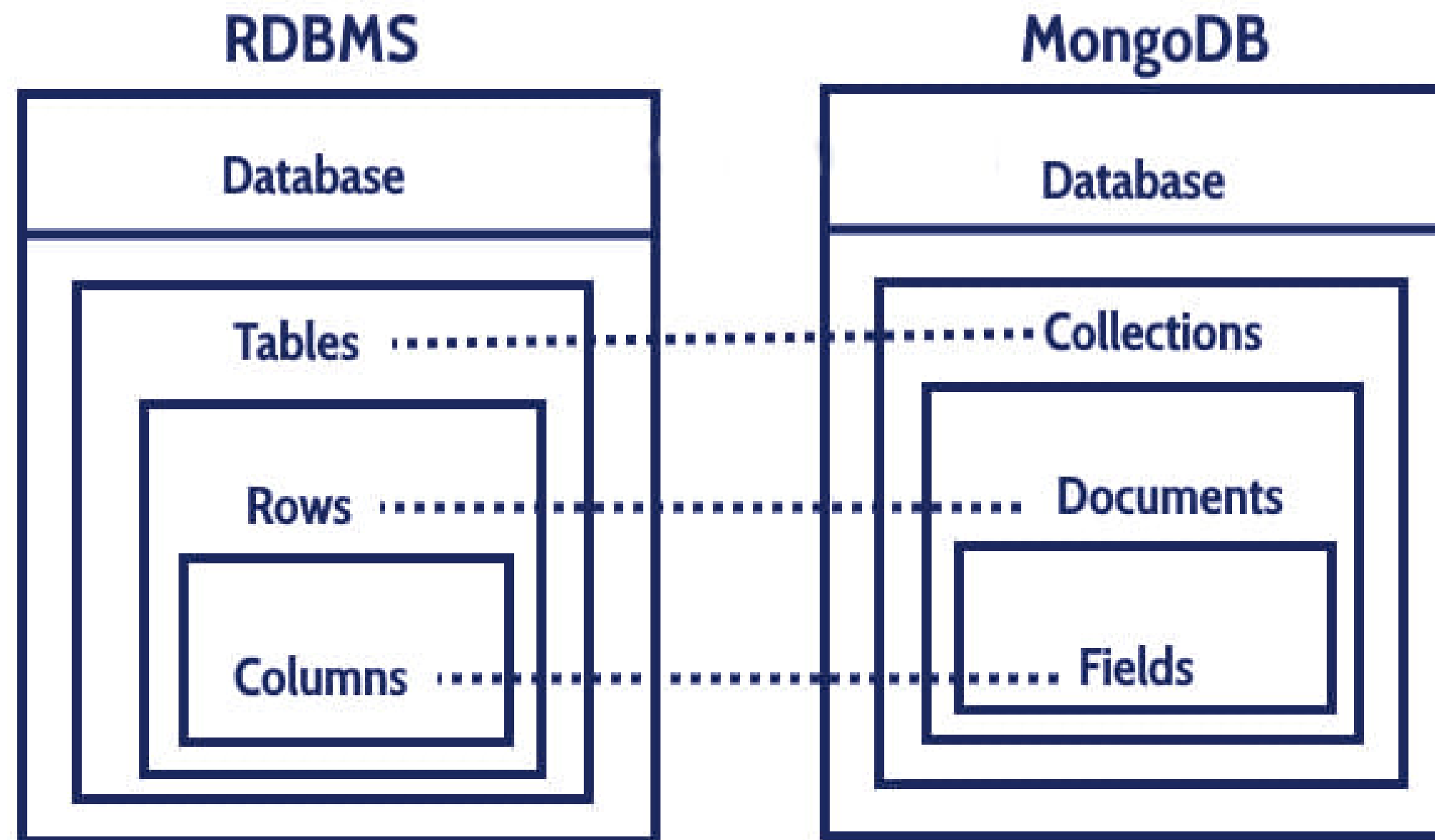
MapReduce

GridFS



mongoDB

Mongo DB 특징



Mongo DB 특징

```
_id: ObjectId("6268e113dc2f9a52ef2d9766")
pk: 414906
model: "movies.movie"
✓ fields: Object
  title: "더 배트맨"
  original_title: "The Batman"
  original_lang: "en"
  popularity: 21029.927
  vote_count: 3583
  vote_average: 7.9
  overview: "지난 2년 간 고담시의 어둠 속에서 범법자들을 응징하며 배트맨으로 살아온 브루스 웨인. 알프레드와 제임스 고든 경위의 도움 아..."
  release_date: "2022-03-01"
  poster_path: "/bCz71ysciwNL2xddSm25ufrgZ7V.jpg"
  backdrop_path: "/5P8SmMzSNYikXpxil6BYzJ16611.jpg"
  youtube_path: "1fAA87ds32o"
  > genres: Array
```

Database Failure

트랜잭션 장애 : 트랜잭션 수행 중 오류로 인한 장애

시스템 장애 : 하드웨어, 소프트웨어 고장으로 인한 장애

디스크 장애 : 디스크 스토리지 일부 혹은 전체 붕괴로 인한 장애

Database Recovery

로그 기반 회복 기법 - 지연갱신 회복 기법, 즉시갱신 회복 기

검사적 회복 기법 - 검사점 이전 처리 회복 제외

그림자 페이징 회복 기법 - 메모리와 하드디스크 테이블 이용

미디어 회복 기법 - 비휘발성 저장 장치가 손상되는 장애 발생을 대비

ARIES 회복 기법 - REDO 중 Repeating history, UNDO 중 Logging

'Transaction' 이란?

데이터베이스에서 데이터의 논리적인 실행 단위

게시판에 글 작성 -> 자신의 글 업데이트 (insert + select : 하나의 transaction)

Transaction 특징

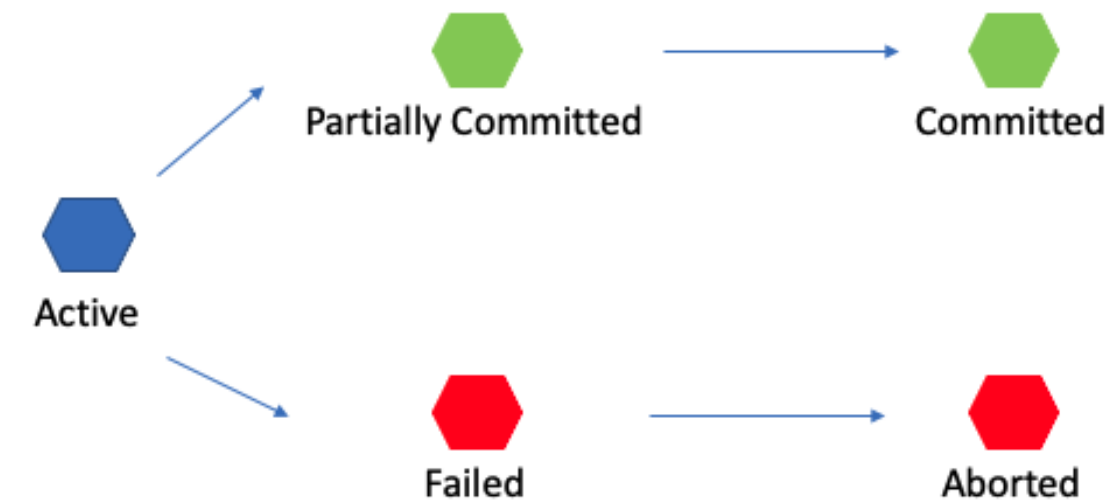
원자성 (Atomicity)

일관성 (Consistency)

독립성 (Isolation)

지속성 (Durability)

트랜잭션의 상태



Q. 무결성을 유지하려는 이유는?

A. 무결성이 유지가 되어야 DB에 저장된 데이터 값과 거기에 해당하는 실제값이 일치하는지 신뢰할 수 있기 때문에

Q. 로킹 단위를 크게했을 때와 작게 했을 때의 차이점은?

. 로킹 단위가 크면 그만큼 관리가 쉽지만 병행성이 떨어진다. 로킹단위가 작으면 관리가 어렵고 오버헤드가 증가하지만, 병행성이 올라간다.

Q. 인덱스를 사용해야 하는 경우는?

A. 데이터의 양이 많고 검색이 변경보다 빈번하거나 필드의 값이 다양한 값을 가지는 경우 인덱스를 사용하여 테이블에 대한 동작의 속도를 높여 줄 수 있다.

Q. 정규화의 목적은?

A. 데이터의 중복을 최소화하고 테이블의 삽입, 삭제, 갱신 과정에서 발생하는 이상현상을 방지하기 위함.

Q. 데이터베이스의 뷰란?

A. 허용된 데이터를 제한적으로 보여주기 위해 하나 이상의 테이블에서 유도된 가상 테이블.

Q. 이상현상이란?

A. 릴레이션에서 일부 속성들의 종속으로 인해 데이터의 중복이 발생하는 것으로 삽입이상, 삭제이상, 갱신이상이 있다.

감사합니다