

# 빅데이터

# 목 차

- 1 빅데이터 저장 계획 수립하기
- 2 빅데이터 저장 모델 설계하기
- 3 빅데이터 저장 관리 시스템 구성하기
- 4 빅데이터 적재하기
- 5 빅데이터 운영하기

# 목 차

## 1 빅데이터 저장 계획 수립하기

1-1 파일 시스템 방식과 관계형 데이터 베이스 시스템

1-2 key-value vs column-oriented 데이터 베이스

1-3 인메모리(in Memory) 방식

1-4 데이터베이스 트랜잭션(Transaction)

2 빅데이터 저장 모델 설계하기

3 빅데이터 저장 관리 시스템 구성하기

4 빅데이터 적재하기

5 빅데이터 운영하기

# 1-1 파일 시스템 방식과 관계형 데이터 베이스 시스템

## File System 방식

빅데이터를 확장 가능한 분산 파일의 형태로 저장하는 방식이다.

**장점 :** 저 사양 서버를 이용하여 대용량, 분산, 데이터 집중형의 애플리케이션을 지원한다.

아파치(Apache)

HDFS(Hadoop Distributed File System)

구글의

GFS(Google File System)

# 1-1 파일 시스템 방식과 관계형 데이터 베이스 시스템

## 관계형 데이터 베이스 시스템

키(key)와 값(value)들의 간단한 관계를 테이블화 시킨 간단한 원칙의 데이터 베이스

관계는 테이블 간에 둘 다 존재.

오라클 DBMS(Oracle)

MS : MS SQL Server

MySQL

Altibase DB

# 1-1 파일 시스템 방식과 관계형 데이터 베이스 시스템

## NoSQL

- A. 스키마가 없다.
- B. 분산처리를 제공한다.

컬럼 베이스 : Hbase

도큐먼트 : MongoDB

키 값 : Redis

그래프 : Neo4J

# 1-2 key-value vs column-oriented 데이터 베이스

## Key-Value 데이터 베이스

- A. 데이터와 키와 그에 해당하는 값의 쌍으로 저장한다.
- B. RDBMS(관계형 데이터 베이스)보다 확장성이 뛰어나고 질의 응답 시간도 빠르다.

# 1-2 key-value vs column-oriented 데이터 베이스

## Column-oriented 데이터 베이스

- A. 로우(row)가 아닌 칼럼(column)기반으로 저장하고 처리한다.
- B. 확장성을 보장하기 위해 여러 개의 노드로 분할되어 저장되고 관리된다.
- C. **Cassandra, Hbase**, HyperTable등



# 1-2 key-value vs column-oriented 데이터 베이스

## document 데이터 베이스

- A. 문서형식의 정보를 저장, 검색, 관리하기 위한 데이터 베이스이다.
- B. **MongoDB**, SimpleDB, couchDB

## 1-3 인메모리(in Memory) 방식

### In-Memory Database (IMDB or MMDB(Main Memory DBMS))

- A. 디스크가 아닌 주 메모리에 모든 데이터베이스를 보유하고 있는 데이터 베이스
- B. [장점] 디스크 검색보다 **자료 접근이 훨씬 빠르다.**
- C. [단점] DB 서버 전원이 갑자기 꺼지면 **자료들이 초고속 즉시 삭제됨.**

# 1-3 인메모리(in Memory) 방식

## In-Memory Database (IMDB or MMDB(Main Memory DBMS))

- A. Altibase(알티베이스) – 한국
- B. SunDB – 선재소프트(한국)
- C. Redis – 오픈소스(2009)
- D. SQLite – Open Source(In memory 지원)

# 1-4 데이터 베이스 트랜잭션(Transaction)

## Database Transaction(데이터베이스 트랜잭션)

- A. 데이터베이스 관리 시스템 또는 유사 시스템에서 상호작용의 단위이다.
- B. (명사) 처리, 처리과정이라 한다.
- C. 다수의 쿼리문으로 이루어진 논리적 작업 단위를 말한다.
- D. 논리적 작업 단위가 모두 성공적으로 실행되면 COMMIT 을 호출하여 모든 쿼리문을 한번에 데이터 베이스에 적용

# 목 차

1 빅데이터 저장, 계획 수립하기

## 2 빅데이터 저장 모델 설계하기

2-1 RDB 저장 시스템 vs NoSQL 저장 시스템

2-2 NoSQL DB 종류

2-3 레코드 식별(기본키), NoSQL DB(id)

2-4 조인(join) 연산

3 빅데이터 저장 관리 시스템 구성하기

4 빅데이터 적재하기

5 빅데이터 운영하기

## 2-1 RDB and NoSQL 저장 시스템

### RDB(관계형 DB), NoSQL DB

#### A. RDB 저장 시스템

Oracle, MSSQL, MySQL, Sybase 등

⇒ 정형 데이터, 반정형 데이터

#### B. NoSQL 저장 시스템 (Not Only SQL)

MongoDB, Cassandra, Hbase, Redis 등

=> 반정형 데이터, 비정형 데이터

## 2-1 빅데이터 기술 유형에 따른 적용예

### 기술 유형과 적용 예

기술 유형	기능 요구사항 예시
연관규칙 마이닝	A 상품을 구매하는 고객이 B상품을 같이 구매하는지 확인할 수 있어야 한다.
분류 분석	특정 고객이 어떤 특성을 갖는 고객 집단에 속하는지 확인할 수 있어야 한다.
머신러닝	기존 영화 구매 패턴을 분석하여 고객에서 특정 영화를 추천할 수 있다.
감정분석	신규 상품에 대한 고객의 반응을 분석할 수 있어야 한다.
회귀분석	구매자의 연령대와 구매 상품의 종류와 특성에 어떤 영향을 미치는지 분석할 수 있어야 한다.
소셜 네트워크 분석	고객사의 소셜 네트워크 상에서의 레퓨테이션은 어떠한지 상시로 모니터링할 수 있어야 한다.
유전 알고리즘	응급실에서 응급처치 프로세스를 어떻게 배치하는 것이 가장 효율적인지 추천할 수 있어야 한다.

## 2-2 NoSQL DB 종류

가. key-value

나. column-oriented

다. document-oriented

라. graph-oriented



## 2-3 레코드 식별(기본키), NoSQL DB(id)

### 가. RDBMS 의 테이블의 레코드 식별

A. Primary key(기본키)

### 나. NoSQL DBMS의 데이터 식별

A. Id 이용

## 2-4 조인(join) 연산

### 가. RDBMS 의 테이블 조인(join)

#### A. Join 연산

### 나. NoSQL DBMS의 조인

#### A. 다른 문서와 연결된 개체(id\_)

#### B. 관계 표현 또는 조인 연산을 위해 Linking, Embedding 기법 이용

## 2-5 MongoDB Data

### ▶ 데이터 삽입 그리고 확인

**cars** 의 **Collection** 에 데이터 넣기, 확인

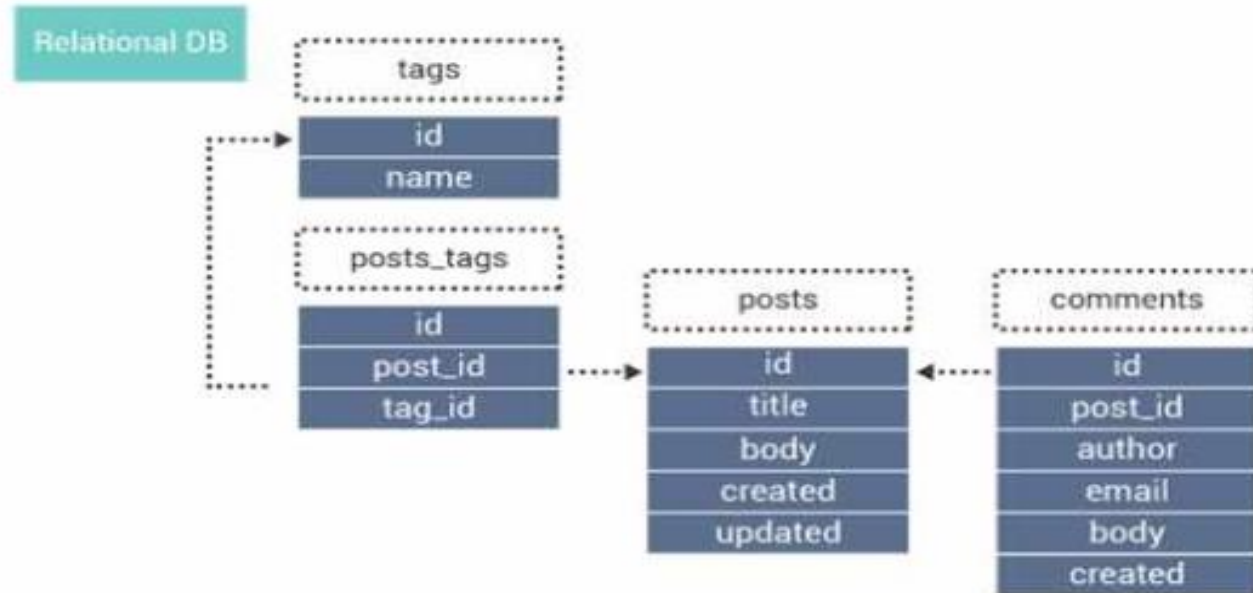
---

```
> db.cars.insert( {  
  name : 'honda',  
  make : 'accord',  
  year : '2010'  
})
```

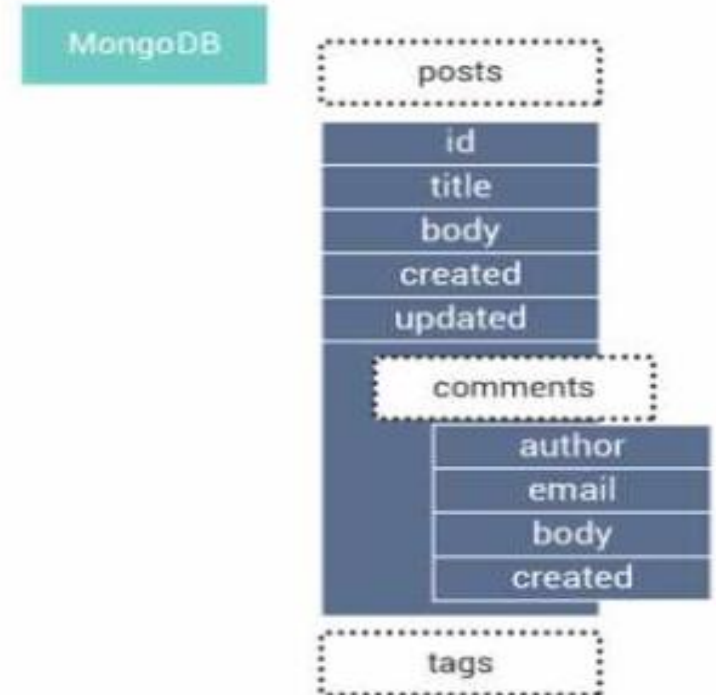
```
> db.car.find().pretty()  
{  
  "_id" : ObjectId("59a6eab6f862e90eeda9093f"),  
  "name" : "honda",  
  "make" : "accord",  
  "year" : "2010"  
}
```

## 2-6 RDBMS vs MongoDB

### ▶ RDBMS vs MongoDB



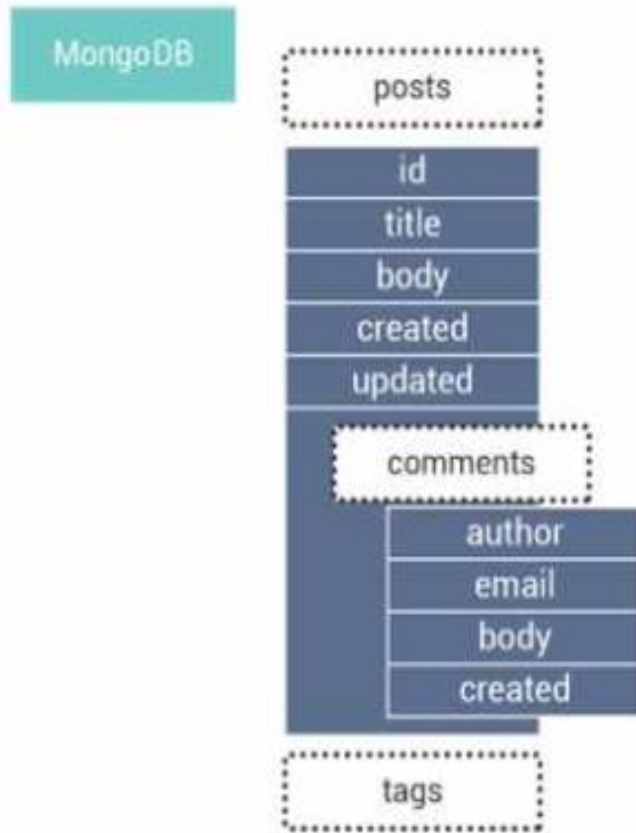
(b) MongoDB 데이터 모델 예



[그림 2-2] 블로그 데이터의 데이터 모델링 사례

## 2-6 RDBMS vs MongoDB

### ► RDBMS vs MongoDB



```
{
  "_id" : ObjectId("4c03e856e258c2701930c091"),
  "title" : "welcome to MongoDB",
  "slug" : "welcome-to-mongodb",
  "body" : "Today, we're gonna totally rock your world...",
  "published" : true,
  "created" : "Mon May 31 2010 12:48:22 GMT-0400 (EDT)",
  "updated" : "Mon May 31 2010 12:48:22 GMT-0400 (EDT)",
  "comments" : [
    {
      "author" : "Bob",
      "email" : "bob@example.com",
      "body" : "My mind has been totally blown!",
      "created" : "Mon May 31 2010 12:48:22 GMT-0400 (EDT)"
    }
  ],
  "tags" : [
    "databases", "MongoDB", "awesome"
  ]
}
```