



# DB Basic - DB란 무엇인가?

CodeSquad Master  
Hoyoung Jung

Data, Database, and DBMS?

Data, Information, and Knowledge?

## 주소록을 개발해 봅시다.

- 어떤 기능이 꼭 필요할까?
- 각 기능의 시간/ 공간 복잡도는?
- 내가 다 구현할 수 있을까?

## 왜 DBMS를 사용할까요?

- 직접 만들기 힘들니까

## 만약에 DBMS를 내가 만든다면?

- 힘들어요. 쓸 데도 없고...

# RDMBS

Relational Database Management System?

E.F. Codd, "A Relational Model of Data for Large Shared Data Banks"

- IBM System R
- Berkeley Postgres

# SQL이란?

- Structured Query Language
- 선언적 언어 cf. 절차적 언어
- RDMBS 제어에 사용됨

## 관계형 데이터베이스

- 오라클, MySQL, MS-SQL, PostgreSQL, ...

# NOSQL

- Not Only SQL
- 관계형 데이터베이스가 아닌 나머지
- 그래프 디비, 도큐먼트 디비, KV Store
- MongoDB, Redis, DyanmoDB, neo4j, Casandra, HBASE, ...

# 데이터베이스 관련 용어

## CAP

- Consistency, Availability, Partition Tolerance

## ACID

- Atomicity, Consistency, Isolation, Durability

## 확장성

- 수평 확장 VS 수직 확장



# SQL VS NoSQL

CAP?

ACID?

Transaction?

**확장성?**

<http://db-engines.com/en/ranking>

# Graph DB

## Neo4j

<https://neo4j.com/why-graph-databases/>

- 복잡한 관계 처리 지원
- 친구 추천 알고리즘, SNS 등에 적합

# Document DB

## MongoDB

<https://www.mongodb.com/>

- json 기반 레코드 저장
- 아무거나 막 저장
- 비교적 빠르고 사용이 간단하다.

### 단점

- 조인에 부적합
- 이전 버전의 경우 쓰기 성능에 문제가 있음

# Key-Value DB

Redis, Mem-cached

- 엄청 빠르다.
- 단순 데이터 처리
- 캐시 / 데이터 분산용

# Cloud DB

## AWS DyanmoDB

- KV Store + Document Store
- 설치가 필요없음
- 고성능, 고가용성, 내구도
- 저렴하지 않은 비용

# ETC

## Hadoop Eco System

- HBASE, Spark, ...

## Cassandra

## Elastic Search