#1. 몽고DB 소개, 기초, 특징

1. NoSQL이란?

기존 관계형DB에서 사용하던 SQL을 사용하지 않는 형식의 DB를 이야기 한다.

NoSQL을 간단하게 분류하면 다음과 같은 4개의 분류 항목을 둘 수 있다.

- 키-값 스토어 : Redis, Dynamo
- 컬럼 지향 스토어 : HBase, Cassandra
- 그래프 DB : Neo4J
- 도큐먼트 지향 스토어 : Mongodb

2. 몽고DB의 장점

1) 친숙함과 이용의 편리성
- 개발자 친화적인 JavaScript를 활용
- JSON과 유사한 형태로 저장하는 방식을 채용
- 배열을 바꾸어서 구조를 간단하게 변경 가능

{id : 1, 이름:'dotdot' , 번호 :[4,3]}
{id : 2, 이름:'rara' , 번호 :5}

- 구조 변경이 RDBMS에 비해 편하다보니 크기를 미리 정하지 않아도 된다 (Schemaless)

-> 최종 개발 명세가 있지 않은 환경에서도 쉽게 사용 할 수 있다.

2) 쉽고 빠른 분산 컴퓨팅 환경 구성

Replicate & Sharding 을 인지하고 설계가되었기 때문에 해당 기능에 유리하다.
커뮤니티버전에도 이 기능을 제공하기에 유지보수에 유리하다. (고가용성, 성능향상)

3. 구조

몽고디비는 DB안에 컬렉션 -> 컬렉션 안에 도큐먼트의 구조로 되어있다.
예시) 회원 정보, 게시글에 대한 정보를 저장 할 때 컬렉션이 없다면
하나의 데이터베이스의 다른 종류의 두 정보가 저장되어 복잡하게 된다.
이럴 때 컬렉션을 통해 종류대로 묶는 역할을 한다. 디비또한 동일하다.

- 도큐먼트 : BSON 구조로 되어있다. BSON 으로 저장되기때문에 데이터 타입의 구조 상 16mb 를 초과할 수 없다. 도큐먼트 하나당 중괄호로 감싸져있다.
- 컬렉션 : 몽고DB Document 의 그룹이며 RDBMS 의 예를 들면 Table 과 개념과 유사합니다.

컬렉션은 단일 데이터베이스에 존재합니다. 컬렉션은 스키마를 강요하지 않습니다. 따라서 컬렉션 내부의 도큐먼트는 서로 다른 필드를 가질수 있습니다. 컬렉션 안에 도큐먼트는 일반적으로 서로 유사한 하거나 관련된 목적이 있습니다.