**데이터베이스 설계서**

Noty 데이터베이스 설계서

2019.09.10

이근혁

1. 데이터 도메인 모델링

* NoSQL 데이터베이스 설계의 경우 RDMS와는 다른 방식으로 접근 해야한다.
* 개체관의 관계보다 쿼리 결과를 지향하고, 데이터 정규화 대신 비정규화를 지향한다.
* (참고 자료: <https://bcho.tistory.com/665>)

|  |
| --- |
|  |

[ Noty 데이터 도메인 모델링 ]

* 각 유저별로 노트, 할 일, 태그 정보를 소유함
* TODO는 Tag의 정보를 가지고 있음

1. 쿼리 결과 디자인

|  |
| --- |
|  |

[ Noty 데이터 쿼리 결과 디자인 ]

1. 태그 정보의 경우 태그명과 태그 색상이 필요로 하며 할 일 데이터는 태그에 따라 분류된다.
2. 할 일 데이터는 진행중/완료 여부와 Note에 기록할 수 있는 번호, 할 일 내용과 어떤 태그에 포함되었는지에 대한 데이터 필요
3. 노트 정보는 노트의 제목과 작성일(또는 수정일), 노트 내용이 필요
4. 데이터 모델링

|  |
| --- |
|  |

[ Noty 데이터 모델 ]

* (User 모델은 제외함 – 도메인 모델과 동일)
* 키 값은 여러 개체의 키 값을 :와 같은 구분자로 이어서 저장 (NoSQL은 RDBMS의 저장방식과 전혀 다르기 때문에 이러한 방식의 비정규화를 지향하는 방향으로 데이터를 설계 해야한다)

1. Prisma 데이터 모델링

* 앞서 진행한 데이터 모델링의 경우 Prisma와 같은 프레임워크 없이 직접 설계할 때 고려할 사항이며 Prisma를 사용하여 모델링 및 배포할 경우 모델링 최적화 과정을 자동으로 처리해준다.
* 앞서 파악한 데이터 간의 연관 관계를 Prisma 모델로 정의하고 배포 진행

|  |
| --- |
|  |

|  |
| --- |
|  |

[ 배포 후 생성된 리소스 ]

* “prisma deploy” 명령어를 입력하여 데이터베이스 생성 및 모델 배포를 자동화할 수 있음
* 배포 시 Docker-compose로 Prisma와 DBMS의 도커 컨테이너가 하나의 서비스로 묶여 배포됨
* 배포 성공 시 개발에 사용할 Prisma 클라이언트 코드가 자동으로 생성되며 해당 코드를 참조하여 데이터베이스의 CRUD 작업 진행