

BIT BY BIT

2025/02/14 (금)

Session 11 – 데이터베이스 Part3

박설진

타임라인

10분 - 오프닝

1시간 - 공유 세션 및 토의 진행

10분 - 마무리

오프닝

강의 듣는데 다들 어떠셨나요?

Ex.

다 들었는지,,

이해는 어느 정도 됐는지,,

시간이 부족하거나 남지는 않았는지 등
숙제여부 / 공부 내용 난이도 / 컨디션

공유 세션

✓ 데이터베이스 정규화과정 #1. 개념과 이상현상 ★★★	06:10
✓ 데이터베이스 정규화과정 #2. 함수적 종속성 ★★★	08:25
✓ 데이터베이스 정규화과정 #3. 제 1정규형 ★★★	01:32
✓ 데이터베이스 정규화과정 #4. 제 2정규형 ★★★	05:23
✓ 데이터베이스 정규화과정 #5. 제 3정규형 ★★★	05:35
✓ 데이터베이스 정규화과정 #6. 보이스 / 코드 정규형 ★★★	04:27
✓ Q. 정규화과정은 꼭 필요한가요? ★★★	03:44
✓ Q. 관계형데이터베이스와 NoSQL데이터베이스의 차이는? ★☆☆	03:19
✓ DEEP DIVE : MongoDB를 쓸 때 주의할 점 ★☆☆	09:56
✓ Q. MySQL의 innoDB, MyISAM 스토리지엔진의 차이는? ★☆☆	04:02
✓ Q. 데이터베이스의 데드락과 해결방법을 설명해주세요. #1 ★★★	04:16
✓ Q. 데이터베이스의 데드락과 해결방법을 설명해주세요. #2 ★★★	05:51

1. 정규화

1. 정규화란?
2. 이상현상이란?

2. 함수적 종속성

1. 함수적 종속성이란? $X \rightarrow Y$
2. 완전함수종속과 부분함수종속
 $\{\text{아이디}\} \rightarrow \text{이름}$
3. $\{\text{아이디}, \text{이벤트번호}\} \rightarrow \text{이름}$

3. 정규화 과정

1. 제 1정규형
2. 제 2정규형
3. 제 3정규형
4. 보이스/코드(BCNF) 정규형
5. 정규화 과정은 왜 필요할까?

3. 정규화과정 #제1정규형

변경전

아이디	이벤트번호	당첨여부	이름	등급
zagabi	221006-1, 221006-2, 221006-3	Y, N, Y	주홍철	골드

3. 정규화과정 #제1정규형

변경후 (앞서 설명한 이상현상이 발생할 수도 있습니다.)

아이디	이벤트번호	당첨여부	이름	등급
zagabi	221006-1	Y	주홍철	골드
zagabi	221006-2	N	주홍철	골드
zagabi	221006-3	Y	주홍철	골드

3. 정규화과정 #제2정규형

아이디	이벤트번호	당첨여부	이름	등급	할인율
zagabi	221006-1	Y	주홍철	골드	3%
zagabi	221006-2	N	주홍철	골드	3%
zagabi	221006-3	Y	주홍철	골드	3%
jong2	221006-4	N	박종선	플레티넘	4%
jong2	221006-5	Y	박종선	플레티넘	4%

3. 정규화과정 #제2정규형

아이디	이벤트번호	당첨여부
zagabi	221006-1	Y
zagabi	221006-2	N
zagabi	221006-3	Y
jong2	221006-4	N
jong2	221006-5	Y

아이디	이름	등급	할인율
zagabi	주홍철	골드	3%
jong2	박종선	플레티넘	4%

3. 정규화과정 #제3정규형

아이디	등급	할인율
zagabi	골드	3%
jong2	플레티넘	4%
justjerk	다이아몬드	5%

3. 정규화과정 #제3정규형

고객 릴레이션

아이디	등급
zagabi	골드
jong2	플레티넘
justjerk	다이아몬드

고객 등급 릴레이션

등급	할인율
골드	3%
플레티넘	4%
다이아몬드	5%

3. 정규화과정 #BCNF 정규형

Boyce-Codd Normal Form

학번	수강명	강사
12010	코딩테스트	큰돌
12010	mevn	재엽
12011	코딩테스트	큰돌
12011	mevn	영주
null	롤	범석

요구 사항은 다음과 같다고 해봅시다.

- 각 수강명에 대해 한 학생은 오직 한 강사의 강의만 수강한다.
- 각 강사는 한 수강명만 담당한다.
- 한 수강명은 여러 강사가 담당할 수 있다.

강사 한명이 인터넷강의 하나만 담당할 수 있고 하나의 인터넷강의는 여러 강사가 담당할 있습니다.

3. 정규화과정 #BCNF 정규형

Boyce-Codd Normal Form

[정규화 과정을 끝낸 테이블]

학번	강사
12010	큰돌
12010	재엽
12011	큰돌
12011	영주

수강명	강사
코딩테스트	큰돌
mevn	재엽
mevn	영주
롤	범석

3. 정규화 과정

도부이결다조
두부...이결..다줘?

정규화 과정 정리

다음 그림과 같습니다.



3. 정규화 과정

정규화 과정 정리

다음 그림과 같습니다.

Q. 정규화 과정은
꼭 필요할까?



4. 관계형 데이터베이스와 NoSQL 데이터베이스의 차이는? 왜?

기능/특징	RDBMS	NoSQL 데이터베이스
스키마	엄격하고 정의되어야 함	유연하고 동적으로 변경 가능
쿼리 언어	SQL	JSON, API, Cypher(Neo4j) 등 다양함. https://neo4j.com/
트랜잭션	지원	지원
격리성(기본)	repeatable_read(mysql)	local := read_uncommitted (mongodb)
스케일링	수직 스케일링이 더 쉬움 - 서버 성능 향상	수평 스케일링이 더 쉬움 - 서버 대수 증가
예	MySQL, Oracle, PostgreSQL	MongoDB, Redis, ElasticSearch, Neo4j

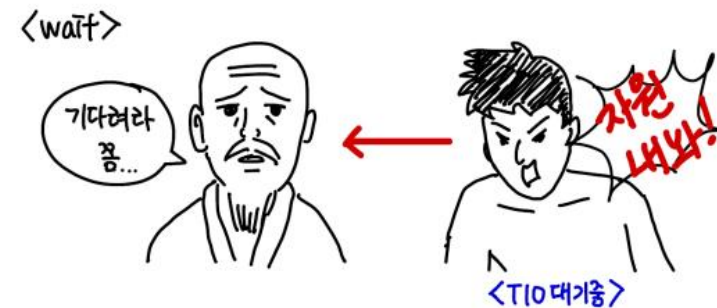
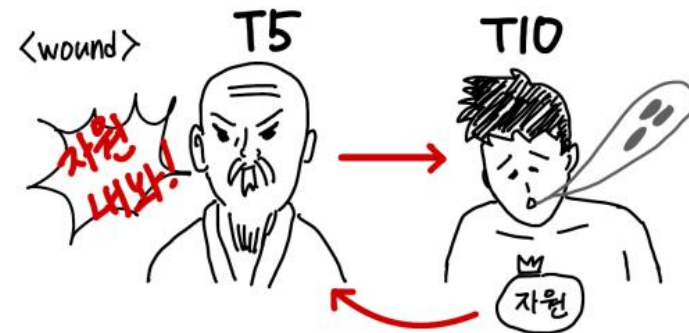
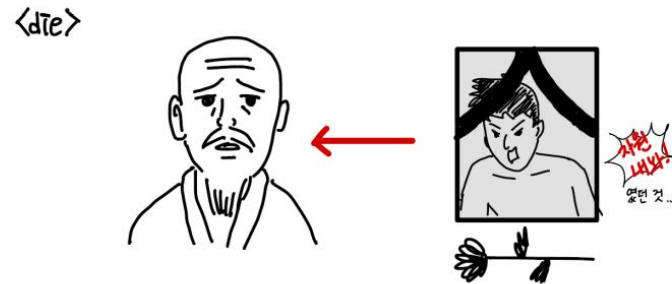
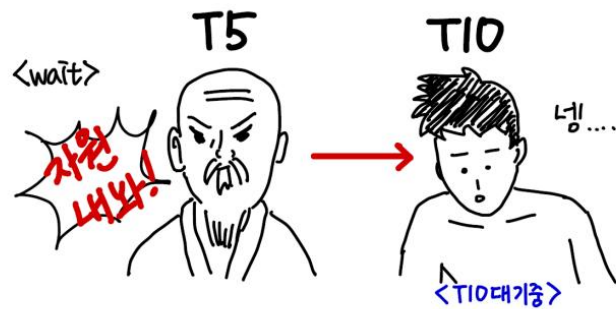
5. 데이터베이스 데드락

1. 데드락이란?
2. 데드락의 발동 조건 네가지
3. 데드락 탐지 방법
4. 데드락 방지 방법

5. 데이터베이스 데드락

1. 데드락이란?
2. 데드락의 발동 조건 네가지
 - 상호배제, 점유대기, 비선점, 순환대기
3. 데드락 탐지 방법
 - timeout, wait for graph
4. 데드락 방지 방법 / (잠금 / 읽기권한)
 - 격리수준 변경, 자원할당 순서 지정, wait-die, wound-wait
 - serializable

5. 데이터베이스 데드락 wait-die, wound-wait (애다운애)



6. 샤딩

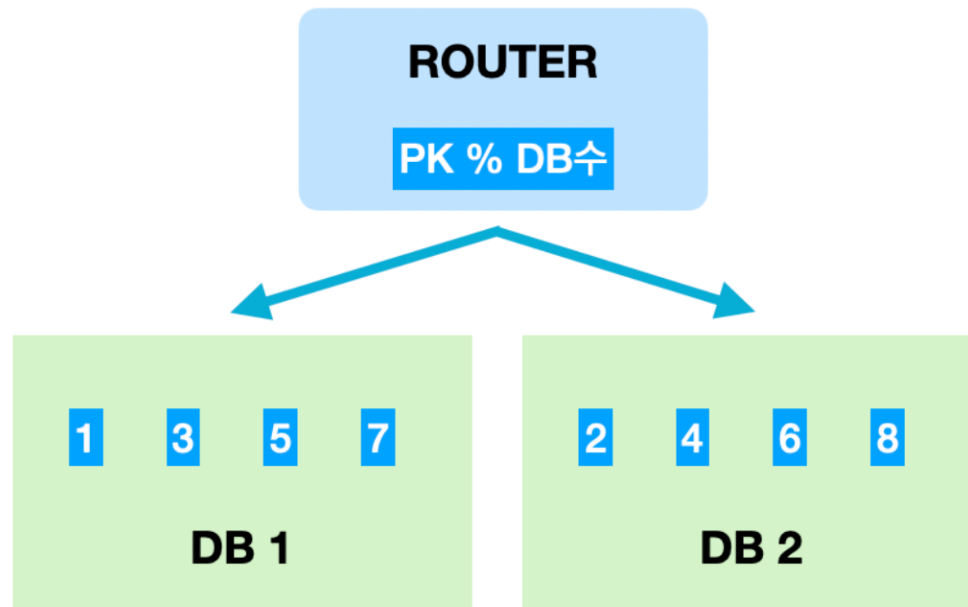
1. 대규모 데이터베이스를 여러 머신에 저장하는 프로세스
2. 데이터를 더 작고 관리 가능한 단위로 수평 분할하는 기법

6. Database Sharding

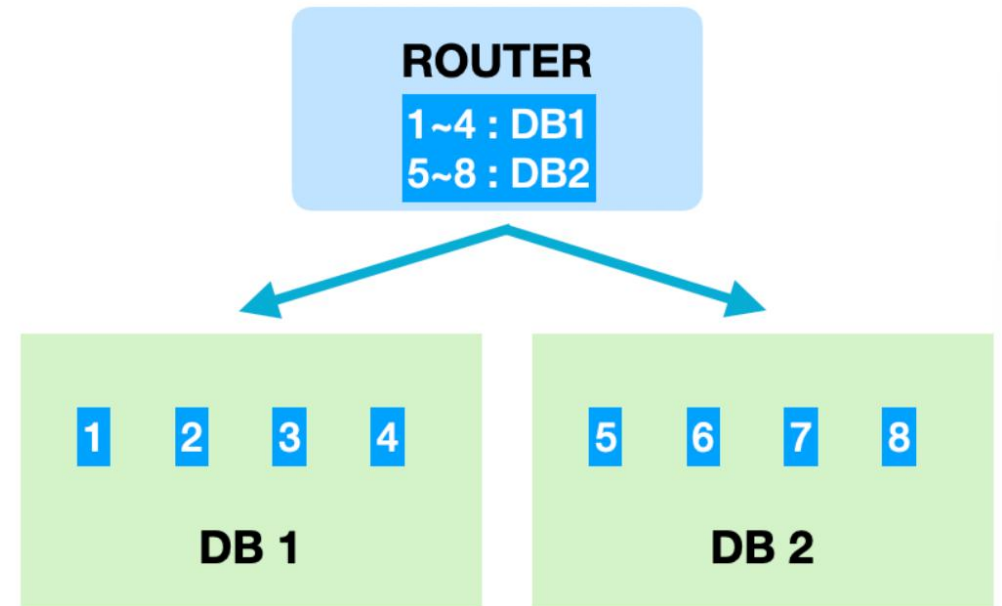
- Hashed sharding: shard key field의 값에 해시를 계산하여 데이터를 분산합니다
 - 이 방식은 데이터가 고르게 분포되지만, 특정 범위의 데이터를 빠르게 찾는 데는 불편할 수 있습니다.
- Ranged sharding: shard key 값에 따라 데이터를 범위로 나누어 분산합니다
 - 이 방식은 범위 검색이 효율적이지만, 데이터가 고르게 분포하지 않으면 특정 샤드에 데이터가 집중되어 불균형이 생길 수 있습니다.
- List sharding : 특정 속성 값에 따라 데이터를 분배하는 방식입니다.
 - 예를 들어, 지역 에 따라 데이터를 분배할 수 있습니다. 이 방식은 고정된 범위에 대한 샤딩에 유리하지만, 데이터가 동적으로 변할 때에는 관리가 어려울 수 있습니다.

6. Database Sharding - range

Modular sharding



Range sharding



6. Database Sharding

샤딩의 장점

- **성능 향상:** 요청을 여러 서버에 분산시켜 처리 속도와 응답 시간을 향상시킬 수 있습니다.
- **확장성:** 새로운 샤드를 추가하여 시스템을 쉽게 확장할 수 있습니다.
- **고가용성:** 여러 서버에 데이터가 분산되어, 특정 서버의 장애가 발생해도 다른 서버에서 처리가 가능해집니다.

샤딩의 단점

- **복잡성:** 데이터를 분산하고 관리하는 시스템이 복잡해지므로, 샤딩을 설정하고 관리하는 데 추가적인 노력과 비용이 필요합니다.
- **쿼리 복잡성:** 여러 샤드에 걸쳐 데이터를 쿼리해야 하는 경우, 성능 저하가 발생할 수 있고 쿼리 작성이 복잡해질 수 있습니다.
- **데이터 일관성 유지:** 분산 시스템에서 데이터 일관성을 유지하는 것이 어려워질 수 있으며, 이를 해결하기 위해 추가적인 작업이 필요합니다.

7. 4NF 5NF

1. 제4정규형(4NF)은 **다중값 종속성**을 제거하여, 한 테이블에서 여러 개의 독립적인 다중값을 저장하지 않도록 합니다.
2. 제5정규형(5NF)은 **조인 종속성**을 처리하여, 관계에서 불필요한 중복을 없애고 데이터를 최소화하여 저장합니다.

7. 4NF : 다중값 종속성 제거

여기 인터넷 서점 회원정보 릴레이션이 있다.

<u>회원번호</u>	이름	주문 도서
121212	김애용	라퐁젤
131313	박꿀꿀	잠자는 숲속의 공주, 백설공주
141414	이멍멍	인어공주, 미녀와 야수

1. 이때
회원번호와 주문도서는 다중값(다치)
종속성 관계 (회원번호->>주문도서)

<u>회원번호</u>	이름	주문 도서
121212	김애용	라퐁젤
131313	박꿀꿀	잠자는 숲속의 공주
131313	박꿀꿀	백설공주
141414	이멍멍	인어공주
141414	이멍멍	미녀와 야수

2. 위 테이블을 쪼갬더니 불필요하게 회원번호,
이름 애트리뷰트가 중복되어 나타남

7. 4NF : 다중값 종속성 제거

4NF 적용 결과

<u>회원번호</u>	이름
121212	김애용
131313	박꿀꿀
141414	이명명

<u>회원번호</u>	주문 도서
121212	라퐁젤
131313	잠자는 숲속의 공주
131313	백설공주
141414	인어공주
141414	미녀와 야수

Q. 다중값종속성 제거 시 키가 결정하는 종속자가 두 개 있고,(회원번호->이름, 회원번호->주문도서) 서로 완전히 독립적이면 쪼개는게 좋다?

A. 서로 완전히 독립적이라면 쪼개는게 바람직하다.
서로 완전히 독립적이지 않다면, (ex. 김애용이 빌릴 수 있는 도서에 제한이 있다면) 이를 분리할 시 정보의 일관성이 깨질 수 있다.

추가 : 다만, 테이블을 분리해도 정보 일관성이 깨지지 않고 이가 더 효율적이라면 분리할 수 있다
(이름-주문도서 테이블을 새로 생성 등)

7. 4NF : 다중값 종속성 제거

하나의 학생이 여러과목, 여러 교수를 가질때
여러과목, 여러교수에 대한 정보를 중복해서 계속 저장함

Student_ID	Club	Hobby
101	축구	독서
101	축구	게임
101	야구	독서
101	야구	게임
102	농구	음악
102	농구	영화

📌 학생 - 동아리 관계 (Student_Club 테이블)

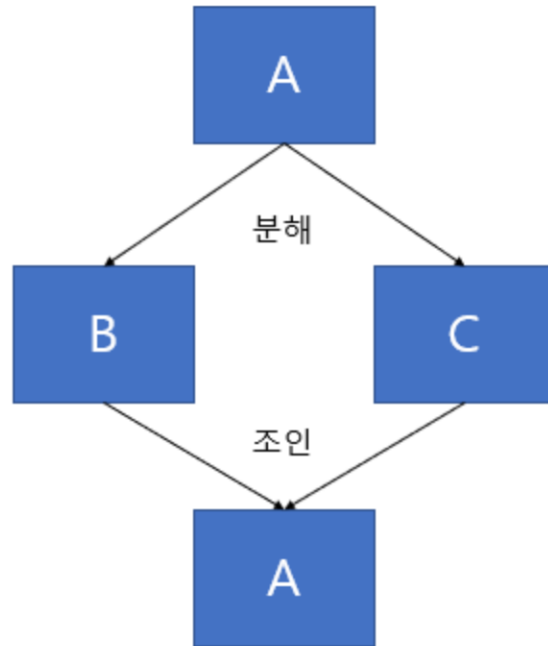
Student_ID	Club
101	축구
101	야구
102	농구

📌 학생 - 취미 관계 (Student_Hobby 테이블)

Student_ID	Hobby
101	독서
101	게임
102	음악
102	영화

7. 5NF : 조인종속성 제거

데이터 중복을 제거하기 위해
더 이상 분해할 수 없을 때 까지 분해



학생번호	기술	언어
S1001	자바	영어
S1001	자바	일본어
S1001	SQL	영어
S1001	SQL	일본어

↓ 조인 종속 제거

학생번호	기술
S1001	자바
S1001	SQL

학생번호	언어
S1001	영어
S1001	일본어

기술	언어
자바	영어
자바	일본어
SQL	영어
SQL	일본어

developer_guide.com

예를 들어 A라는 릴레이션을 B와 C로 분해했다가 다시 조인했을 때 그대도 A가 된다면, A는 조인 종속성이 있다고 한다.

7. 5NF : 조인종속성 제거

세 개 이상의 테이블을 결합할 때 발생하는 불필요한 중복 데이터를 제거
모든 관계가 조인으로 유도될 수 있어야 하며, 불필요한 중복을 없애기 위해 테이블을 추가적으로 분리

학생ID	과목	교수
1	수학	김교수
1	물리	박교수
1	화학	이교수

1. 학생과 과목 관계:

학생ID	과목
1	수학
1	물리
1	화학

2. 과목과 교수 관계:

과목	교수
수학	김교수
물리	박교수
화학	이교수

3. 학생과 교수 관계:

학생ID	교수
1	김교수
1	박교수
1	이교수

감사합니다