

11주차

- 트랜잭션이란 무엇인가요? 4가지 원칙을 포함해서 설명해 주세요.

데이터베이스의 상태를 변경시키기 위해 수행하는 작업 단위
중간에 실행에 문제가 생겼을 경우 롤백하고 오류 없이 완료되면 커밋

ACID(4가지 원칙)

원자성 Atomicity

데이터베이스에 모두 반영되거나 안되거나
롤백과 커밋으로 관리
+)save point: 롤백 시 처음이 아닌 중간 지점부터 다시 작업을 수행하도록 지정

일관성 Consistency

작업의 처리 결과가 항상 일관돼야

독립성 Isolation

각각의 트랜잭션은 서로 독립적으로 영향을 주거나 받으면 안됨
데이터를 읽거나 쓸 때 lock을 걸어서 데이터 정합성 유지
→ 데드락 발생 주의

지속성 Durability

트랜잭션의 결과가 영구적으로 결과에 반영 by 커밋

- 트랜잭션의 격리 수준과 각 수준에서 발생할 수 있는 문제들에 대해 말해보세요.

트랜잭션 격리 수준: 여러 트랜잭션이 병렬로 처리될 때, 한 트랜잭션이 조회, 변경하는 데이터에 대한 다른 트랜잭션들의 조회 허용 여부 수준
데이터 정합성과 성능 반비례

- Dirty Read

- Non-Repeatable Read

한 트랜잭션 내에서 같은 데이터 여러 번 조회시 읽어온 데이터가 다름

- Phantom Read

Non-Repeatable Read의 한 종류
조회해서 가져온 결과 행이 새로 생기거나 없어짐

SERIALIZABLE (직렬화 가능)

접근 중인 테이블의 모든 행을 다른 트랜잭션에서 접근 불가

REPEATABLE READ (반복 가능한 읽기)

트랜잭션이 시작되기 전에 커밋된 내용에 대해서만 조회 허용하는 격리수준

- Phantom Read

READ COMMITTED (커밋된 읽기)

커밋된 변경사항만 다른 트랜잭션에서 조회할 수 있도록 허용하는 격리 수준

커밋 이전까지는 이전 데이터 값이 조회되고 커밋 이후에는 변경된 값으로 조회됨

- Phantom Read
- Non-Repeatable Read

READ UNCOMMITTED (커밋되지 않은 읽기)

커밋되지 않은 데이터 변경 내용을 다른 트랜잭션에서 접근하여 조회 가능

- Dirty Read
- Phantom Read
- Non-Repeatable Read

+) 어림잡아 하는 연산에서 사용하기 좋음

+) MySQL의 기본 격리 수준 : REPEATABLE READ

오라클의 기본 격리 수준: READ COMMITTED

- 공유 락과 배타 락의 차이는 무엇인가요? 영선님

데이터의 일관성을 보장하기 위한 방법

공유락 읽기o 쓰기 x

공유락은 동시에 여러개 걸 수 있음

한 트랜잭션이 공유락을 걸면 해당 자원에 배타락 걸 수 없음

배타락 읽기x 쓰기 x

배타락이 걸리면 해당 자원에 배타락, 공유락 걸 수 없음

• 데드락이란 무엇이며, 어떻게 발생할까요?

데드락: 두 개 이상의 트랜잭션이 특정 자원(테이블 또는 행)의 Lock을 획득한 채 다른 트랜잭션이 소유하고 있는 잠금을 요구하면 아무리 기다려도 상황이 바뀌지 않는 상태

+)예방법:

- 각 트랜잭션이 실행되기 전에 필요한 모든 자원을 Lock
→ 병행성이 떨어지는 단점

+)회피법: 자원을 할당할 때 시간 스탬프(Time Stamp)를 활용해서 교착상태가 일어나지 않도록 회피

• Wait-Die 방식

비선점 기법에 기반, 다른 트랜잭션이 데이터를 점유하고 있을 때 기다리거나(Wait) 포기(Die)

- 트랜잭션 A가 트랜잭션 B에 의해 잠금된 데이터를 요청할 때 트랜잭션 A이 먼저 들어온 트랜잭션이라면 대기(Wait)한다.
- 트랜잭션 A가 나중에 들어온 트랜잭션이라면, 포기(Die)하고 나중에 다시 요청한다.

• Wound-Wait 방식

선점 기법, 다른 트랜잭션이 데이터를 점유하고 있을 때 빼앗거나(Wound) 기다리는(Wait) 방식

- 트랜잭션 A가 트랜잭션 B보다 먼저 들어온 트랜잭션이라면, 데이터를 선점(Wound)한다.
- 반면, 트랜잭션 A가 트랜잭션 B보다 나중에 들어온 트랜잭션이라면 대기(Wait)한다