

트랜잭션 / 트랜잭션 격리 수준

트랜잭션 ?

데이터베이스의 상태를 변화 시키기 위해 수행하는 작업의 단위

- 은행 거래가 일어날때의 과정

- A 계좌에서 100만원 출금
- B 계좌에 100만원 입금

- 거래가 일어날 때 실행되는 쿼리

- UPDATE문 : A계좌에서 100만원 빼기
- UPDATE문 : B계좌에 100만원 더하기

A계좌에서는 돈이 빠졌으나, 서버다운 등의 이유로 B계좌에 입금이 되지 않는 상황이 발생할 수 있다. 이러한 상황을 방지하기위해 나온것이 트랜잭션

트랜잭션의 성질

- 트랜잭션이 안전하게 수행된다는 것을 보장하기 위한 성질

ACID

- Atomicity(원자성)

트랜잭션은 DB에 모두 반영되거나, 전혀 반영되지 않아야 한다.(= 완료되지 않은 트랜잭션의 중간 상태를 DB에 반영해서는 안 된다.)

- Consistency(일관성)

트랜잭션 작업처리결과는 항상 일관성 있어야 한다. DB는 항상 일관된 상태로 유지되어야 한다.

- Isolation(독립성)

둘 이상의 트랜잭션이 동시 실행 되고 있을 때, 어떤 트랜잭션도 다른 트랜잭션 연산에 끼어들 수 없다.

- Durability(지속성)

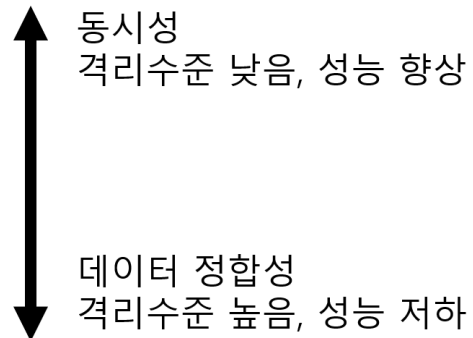
트랜잭션이 성공적으로 완료되었으면 결과는 영구히 반영되어야 한다.

But, ACID는 트랜잭션이 이론적으로 보장해야하는 성질. 실제로는 성능을 위해 성질 규정이 완화.

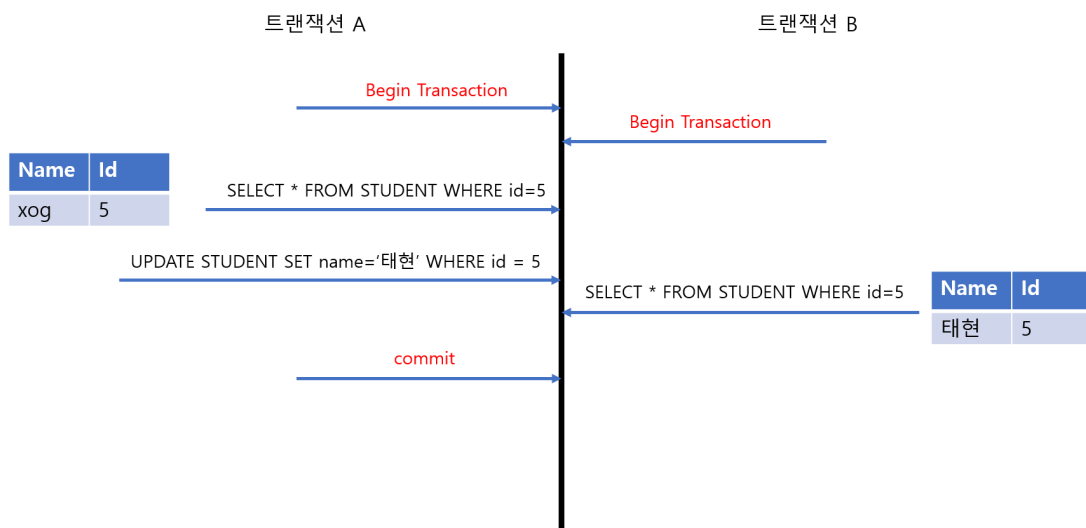
트랜잭션 격리 수준

- 동시에 DB에 접근 할 때 그 접근을 어떻게 제어할지에 대한 설정
- **트랜잭션 수준 읽기 일관성** 을 지키기 위해 있음
 - 트랜잭션이 시작된 시점으로부터 일관성 있게 데이터를 읽어 들이는 것을 말한다.
 - 하나의 트랜잭션이 진행되는 동안 다른 트랜잭션에 의해 변경사항이 발생하더라도 이를 무시하고 계속 일관성 있는 데이터를 보여준다. (물론 트랜잭션 자신이 발생한 변경사항은 읽을 수 있다)

READ-UNCOMMITTED
READ-COMMITTED
REPEATABLE-READ
SERIALIZABLE

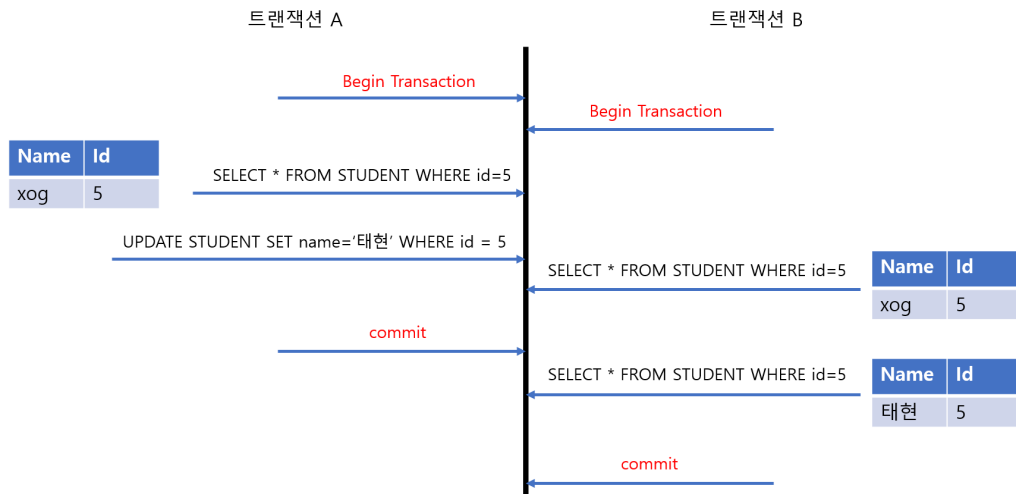


- READ-UNCOMMITTED (레벨 0)
 - 커밋 전의 트랜잭션의 데이터 변경 내용을 다른 트랜잭션이 읽는 것을 허용



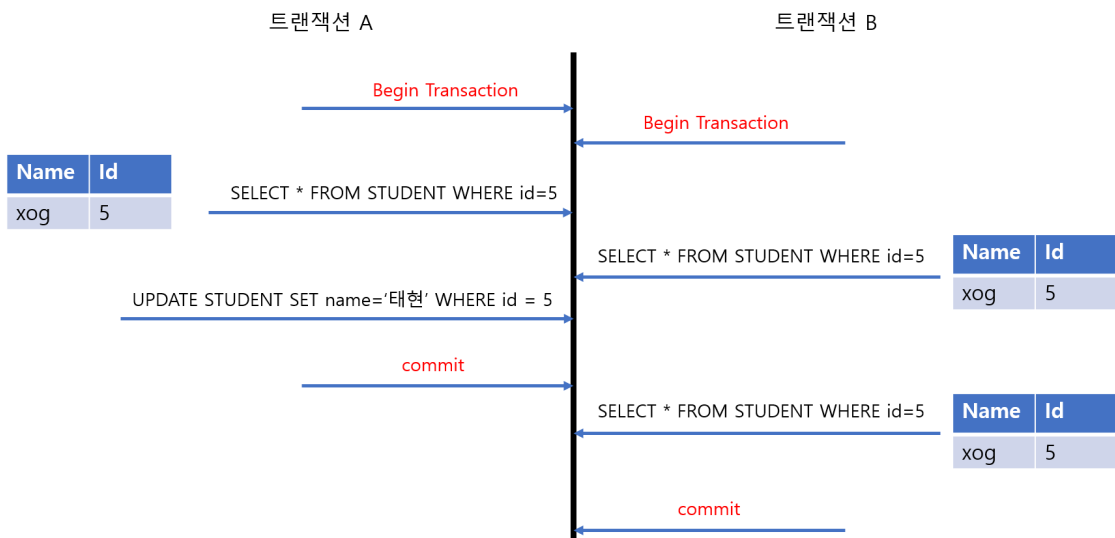
- READ-COMMITTED
 - 커밋이 완료된 트랜잭션의 변경사항만 다른 트랜잭션에서 조회 가능

- SQL 서버가 default로 사용하는 격리 수준



• REPEATABLE-READ

- 트랜잭션이 시작되기 전에 커밋이 완료된 데이터만 읽을 수 있는데, READ-COMMITTED와 다른 점은, 트랜잭션 범위 내에서 조회한 내용이 항상 동일함을 보장



• SERIALIZABLE

- 한 트랜잭션에서 사용하는 데이터를 다른 트랜잭션에서 접근 불가
- 완벽한 읽기 일관성 모드 제공
- 다른 사용자는 트랜잭션 영역에 해당하는 데이터에 대한 수정 및 입력 불가능

격리수준 별 발생하는 문제점

문제 유형 Isolation Level	Dirty Read	Non-Repeatable Read	Phantom Read
Read Uncommitted (Level 0)	가능	가능	가능
Read Committed (Level 1)	불가능	가능	가능
Repeatable Read (Level 2)	불가능	불가능	가능
Serializable (Level 3)	불가능	불가능	불가능

- Dirty Read
 - 커밋되지 않고, 수정중인 데이터를 다른 트랜잭션에서 읽을 수 있도록 허용할 때 발생
 - 어떤 트랜잭션에서 아직 실행이 끝나지 않은 다른 트랜잭션에 의한 변경사항을 보게 됨
- Non-Repeatable Read
 - 한 트랜잭션에서 같은 쿼리를 두 번 수행할 때 그 사이에 다른 트랜잭션 값을 수정 또는 삭제하면서 두 쿼리의 결과가 상이하게 나타나는 일관성이 깨진 현상
- Phantom Read
 - 한 트랜잭션 안에서 일정 범위의 레코드를 두 번 이상 읽었을 때, 첫번째 쿼리에서 없던 레코드가 두번째 쿼리에서 나타나는 현상
 - 트랜잭션 도중 새로운 레코드 삽입을 허용하기 때문에 나타나는 현상

트랜잭션 격리수준 선택 시 고려사항

트랜잭션 격리 수준에 대한 조정은 동시성과 데이터 무결성에 연관되어있다. 동시성을 증가시키면 데이터 무결성에 문제가 발생하고, 데이터 무결성을 유지하면 동시성이 떨어지게 된다.

레벨을 높게 조정할 수록 발생하는 비용이 증가

reference

- <https://velog.io/@guswns3371/데이터베이스-트랜잭션-격리수준>
- <https://www.youtube.com/watch?v=e9PC0sroCzc>
- <https://gyoogle.dev/blog/computer-science/data-base/Transaction Isolation Level.html>