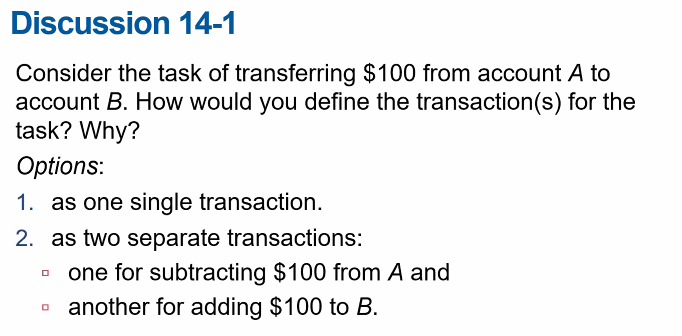
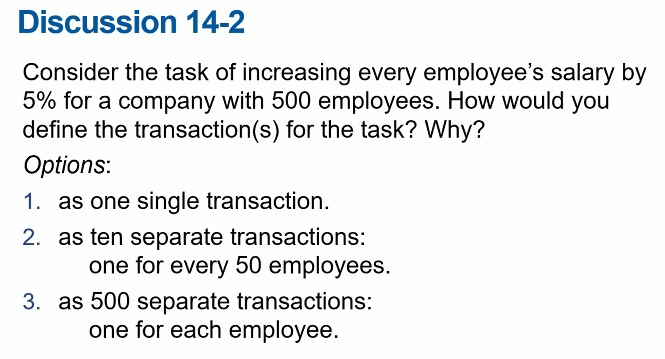
**Discussion 06/03**

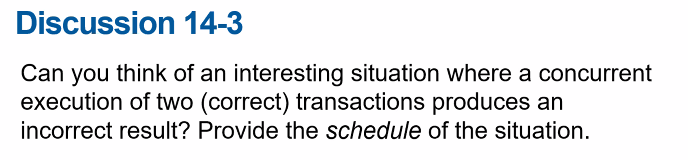


1번을 선택해야 하지 않을까? 이유는 2번을 선택할 경우 A 과정을 commit 후에 B 연산을 처리하는 도중 에러가 발생해 rollback 해야 할 경우 문제가 생김.



3번이 좋지 않을까? 1번과 2번은 에러가 발생하면 너무 많은 연산을 rollback해야 하므로 비효율적.

=> 근데 트랜잭션 개수가 많아질수록 오버헤드가 커짐. 너무 많아도 안 좋다. Fail이 없다면 옵션 1도 괜찮음.



A에 2000원이 있을 때 500원을 넣고 2500원을 뺄 경우

T1

Read A

A = A + 500

T2

Read A

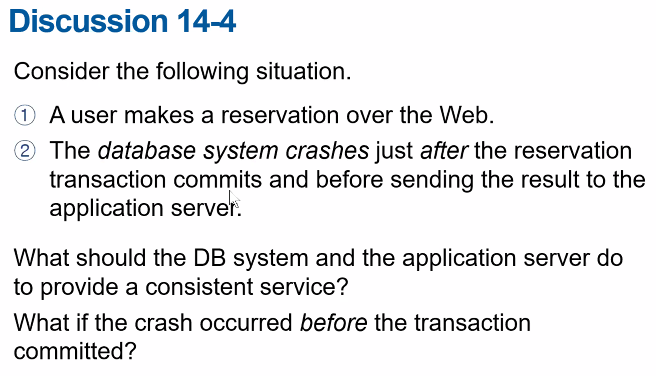
A = A – 2500

Write A

T1

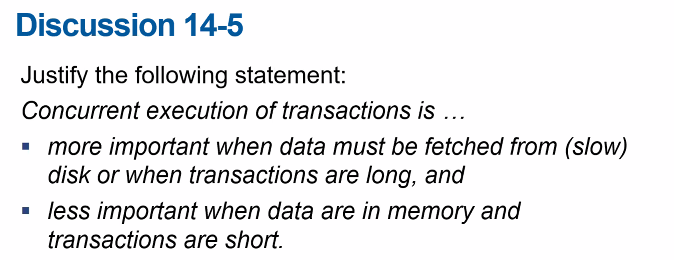
Write A

T1의 결과가 쓰이기 전에 T2에서 Write가 실행되면 A 계좌가 -500이 됨. Invalid.



어플리케이션 서버와 DBMS 각각에 로그나 상태 등이 저장되어 있을 텐데, 데이터베이스를 복구할 때 로그를 읽어보고 트랜잭션이 commit 되어 있는데 application 서버의 상태에 반영이 되어있지 않으면 해당 트랜잭션 결과를 application 서버로 전송해준다.

트랜잭션이 commit되기 전에 crash가 발생하면 해당 트랜잭션을 rollback해서 abort 상태로 만든 뒤 유저에게 오류 발생 알림을 띄워야 할 것 같다.



속도 측면에서 봤을 때 첫 번째 경우에서 동시 실행을 할 수 없다면 각각의 트랜잭션을 처리하는 데 시간이 너무 많이 걸리기 때문에? 두 번째는 각각의 트랜잭션의 수행 시간이 짧기 때문에 덜 중요하다.