

3. 오손판독(dirty read)

시간	T1	T2	수행 과정
	read(x)		x = 100이면
	x = x + 30		x = x + 30 = 100 + 30 = 130
	write(x)		x = 130 기록
		read(x)	x = 130 (dirty data)
		x = x - 50	x = x - 50 = 130 - 50 = 80
		write(x)	x = 80
	rollback		T1은 복귀

- 오손판독은 쓰기-읽기 충돌이라 한다.
- 완료되지 않은 트랜잭션 T1이 갱신 기록한 데이터 x값을 T2가 읽고 갱신 기록하였다.
- 트랜잭션 T1이 최종적으로 정상 완료되지 않고 철회(rollback, 복귀)되면
- 잘못된 데이터를 읽은 트랜잭션 T2도 당연히 복귀되어야 한다.(연쇄복귀)
- 만약, 트랜잭션 T2가 갱신 작업을 성공적으로 완료하고 시스템을 떠난 뒤이면 복귀할 수도 없는 문제가 발생된다.(회복 불능) - 비완료 의존성 문제
- 오손판독은 완료되지 않은 트랜잭션에서 쓴(write) 데이터를 다른 트랜잭션에서 읽는 것
- 트랜잭션들이 동시에 같은 데이터에 접근하여 갱신할 때
 - 어떤 트랜잭션은 실행 성공하여 완료된 상태(복귀가 필요 없음)
 - 다른 트랜잭션은 실행 실패(갱신을 취소하고 원래 상태로 복귀해야 함)
- 다른 트랜잭션이 처리한 부분에 대해서는 취소가 불가능하다.(회복 불능)

// 오손판독 문제점 해결 방법

- 부분 수행 중인 트랜잭션의 데이터 값을 참조하지 않도록 한다.
- 완전히 완료한 트랜잭션의 데이터 값을 참조하면 이와 같은 문제는 발생하지 않는다.
- 트랜잭션 내에서 충돌(conflict)이 발생하면, DBMS는 병행제어 기법을 통해 트랜잭션이 순차적인 실행이 되도록 만들어 주어야 한다.

오손판독	비완료 의존성 문제(uncommitted dependency problem), 연쇄복귀(cascading rollback), 임시 갱신 문제라고도 한다.
------	---

기출문제 분석

1. 다음과 같이 동시성제어 없이 두 트랜잭션 T1과 T2가 수행되는 경우 X의 최종값은? (단, X의 초기값은 100이다) [2015년 국가 7급]

시간	T1	T2
	Read_item(X)	
	$X = X + 30$	
	Write_item(X)	
		Read_item(X)
	Rollback	
		$X = X - 50$
		Write_item(X)

- ① 30 ② 50
- ③ 80 ④ 100

☞ 오손판독(dirty read) 현상 발생

- 오손판독은 완료되지 않은 트랜잭션에서 쓴(write) 데이터를 다른 트랜잭션에서 읽는 것
 - 트랜잭션 T1이 갱신하고 기록한 데이터를 T2가 읽었다.
 - 트랜잭션 T1이 최종적으로 롤백되면 그 값을 읽은 트랜잭션은 모순 상태가 된다.
- 동시성제어 없이 두 트랜잭션 T1과 T2가 수행되는 경우

시간	T1	T2	수행 과정
	Read_item(X)		$X = 100$
	$X = X + 30$		$X = X + 30 = 100 + 30 = 130$
	Write_item(X)		$X = 130$
		Read_item(X)	$X = 130$
	Rollback		T1은 복귀
		$X = X - 50$	$X = X - 50 = 130 - 50 = 80$
		Write_item(X)	$X = 80$

- 만약, 동시성제어가 정상 수행되면 T1은 Rollback되므로 T1이 변경한 값은 취소된다.
 - 해서, $X = X - 50 = 100 - 50 = 50$

2. 2개의 계좌 A1, A2의 잔액이 각각 2,000,000원, 3,000,000원이라고 하자. 이 상태에서 각 계좌의 관리를 수행하는 트랜잭션 T1 및 T2의 실행이 다음과 같을 때, 발생한 현상으로 옳은 것은?
[2018년 국가 7급]

- 1 : T2가 계좌 A2에 2,500,000원을 더해 5,500,000원이 됨
- 2 : T1이 계좌 A1에 1,500,000원을 더해 3,500,000원이 됨
- 3 : T2가 계좌 A1의 잔액(3,500,000원)이 충분하다고 판단하여 A1에서 A2로 2,500,000원 이체를 실행함
- 4 : T1이 알 수 없는 이유로 트랜잭션의 수행이 실패함

- ① 오손판독(dirty read)
- ② 반복할 수 없는 판독(non-repeatable read)
- ③ 부정확한 요약(incorrect summary)
- ④ 유령판독(phantom read)

☞ 오손판독(dirty read)

• 주어진 내용을 분석하면

T1	T2
	read(A2); A2 = A2 + 2,500,000; write(A2);
read(A1); A1 = A1 + 1,500,000; write(A1);	
	read(A1); A1 = A1 - 2,500,000; write(A1); read(A2); A2 = A2 + 2,500,000; write(A2);
rollback(복귀)	

- 트랜잭션 T1이 갱신 기록한 데이터 A1 값을 T2가 읽고 갱신 기록하였다.
- 트랜잭션 T1이 최종적으로 정상 완료되지 않고 철회(rollback, 복귀)되었다.
→ 잘못된 데이터를 읽은 트랜잭션 T2도 당연히 복귀되어야 한다.(오손판독, 연쇄복귀)
- 오손판독은 완료되지 않은 트랜잭션에서 쓴(write) 데이터를 다른 트랜잭션에서 읽는 것

3. <보기>의 트랜잭션의 무제어 동시 공용 사례 (가)와 (나)에서, 각각에 해당하는 문제점을 옳게 짚 지은 것은? [2021년 서울 7급]

---<보기>---

(가)	T1	T2	(나)	T1	T2
	read(x) x ← x + 300 write(x)	read(x) x ← x * 5 write(x)		read(x) x ← x + 300 write(x) read(y) rollback T1	read(x) x ← x * 5 write(x)

- | | |
|----------------------------|--------------------------|
| (가) | (나) |
| ① 갱신분실(lost update) | 모순성(inconsistency) |
| ② 모순성(inconsistency) | 연쇄복귀(cascading rollback) |
| ③ 갱신분실(lost update) | 연쇄복귀(cascading rollback) |
| ④ 연쇄복귀(cascading rollback) | 갱신분실(lost update) |

☞ 트랜잭션의 무제어 동시 공용 사례 - 문제점

갱신분실 쓰기-쓰기 충돌	시간	T1	T2	수행 과정
	↓	read(x)		x = 1000이면
		x = x + 100		x = x + 100 = 1100
			read(x)	x = 1000
			x = x + 200	x = x + 200 = 1200
		write(x)		x = 1100 기록(갱신분실)
		write(x)	x = 1200 기록(T1의 갱신 무효화)	
오손판독 {연쇄복귀} 쓰기-읽기 충돌	시간	T1	T2	수행 과정
	↓	read(x)		x = 100이면
		x = x + 30		x = x + 30 = 100 + 30 = 130
		write(x)		x = 130 기록
			read(x)	x = 130 (dirty data)
			x = x - 50	x = x - 50 = 130 - 50 = 80
			write(x)	x = 80
		rollback		T1은 복귀