

Chap11. 트랜잭션 제어와 세션

- ▶ 11-1 하나의 단위로 데이터를 처리하는 트랜잭션
- ▶ 11-2 트랜잭션을 제어하는 명령어
- ▶ 11-3 세션과 읽기 일관성의 의미
- ▶ 11-4 수정 중인 데이터 접근을 막는 LOCK

11-1 하나의 단위로 데이터를 처리하는 트랜잭션

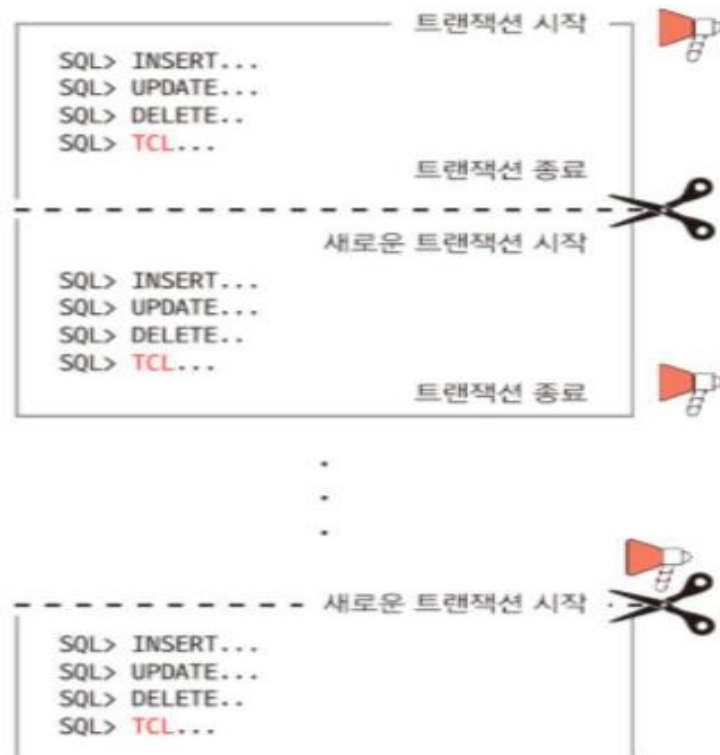
-
- ▶ 트랜잭션이란?
 - ▶ 더 이상 분할할 수 없는 최소 수행 단위
 - ▶ 한 개 이상의 DML
 - ▶ ALL OR NOTHING



11-1 하나의 단위로 데이터를 처리하는 트랜잭션

▶ 트랜잭션이란?

- ▶ 더 이상 분할할 수 없는 최소 수행 단위
- ▶ 한 개 이상의 DML



트랜잭션은 TCL 명령어를 사용하는 시점에 끝나며 새로운 트랜잭션이 시작됩니다.

11-1 하나의 단위로 데이터를 처리하는 트랜잭션

▶ 트랜잭션이란?

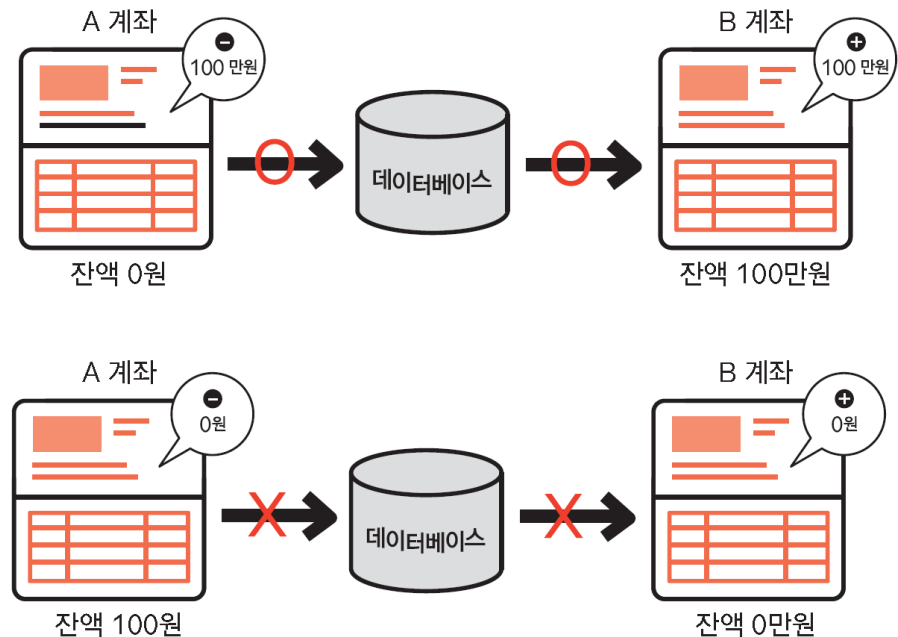
▶ ALL OR NOTHING

① A 계좌 잔액을 0원으로 변경하는 UPDATE문 실행

```
UPDATE ACCOUNT  
  SET BALANCE = 0  
WHERE ACCNO = A계좌번호;
```

② B 계좌 잔액을 100만 원으로 변경하는 UPDATE문 실행

```
UPDATE ACCOUNT  
  SET BALANCE = 1000000  
WHERE ACCNO = B계좌번호;
```

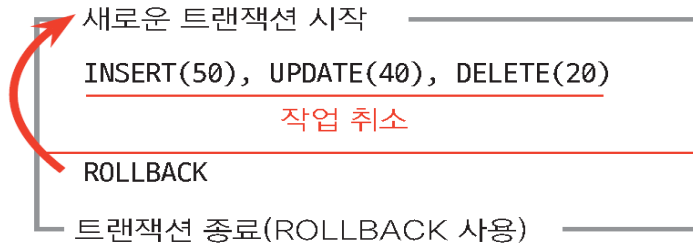


11-1 하나의 단위로 데이터를 처리하는 트랜잭션

- ▶ **TCL(Transaction Control Language)**
 - ▶ 트랜잭션을 제어하기 위해 사용하는 명령어
 - ▶ 트랜잭션을 취소하고 싶을 때는 **ROLLBACK**
 - ▶ 트랜잭션을 영원히 반영하고 싶을 때는 **COMMIT**

11-2 트랜잭션을 제어하는 명령어

▶ 트랜잭션을 취소하고 싶을 때는 **ROLLBACK**



ROLLBACK을 실행하면 현재 트랜잭션이 시작된 시점까지 작업 취소 즉, ROLLBACK 직전에 실행한 INSERT, UPDATE, DELETE 문은 실행 취소됩니다.

실습 11-1 DEPT 테이블을 복사해서 DEPT_TCL 테이블 만들기

```
01 CREATE TABLE DEPT_TCL
02     AS SELECT *
03     FROM DEPT;

04 SELECT * FROM DEPT_TCL;
```

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

실습 11-2 DEPT_TCL 테이블에 데이터를 입력·수정·삭제하기

```
01 INSERT INTO DEPT_TCL VALUES(50, 'DATABASE', 'SEOUL');

02 UPDATE DEPT_TCL SET LOC = 'BUSAN' WHERE DEPTNO = 40;

03 DELETE FROM DEPT_TCL WHERE DNAME = 'RESEARCH';

04 SELECT * FROM DEPT_TCL;
```

실습 11-3 ROLLBACK으로 명령어 실행 취소하기

```
01 ROLLBACK;

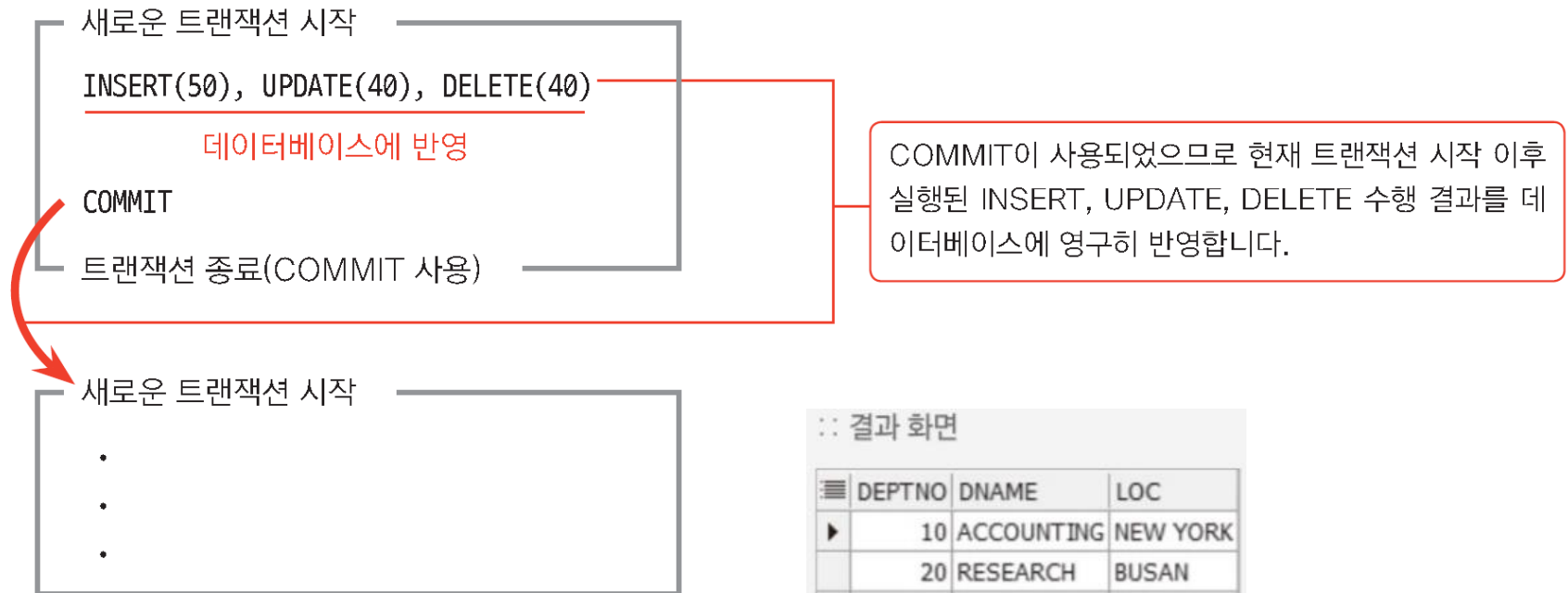
02 SELECT * FROM DEPT_TCL;
```

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

11-2 트랜잭션을 제어하는 명령어

▶ 트랜잭션을 영원히 반영하고 싶을 때는 COMMIT



실습 11-4 DEPT_TCL 테이블에 데이터를 입력·수정·삭제하기

```
01 INSERT INTO DEPT_TCL VALUES(50, 'NETWORK', 'SEOUL');
02 UPDATE DEPT_TCL SET LOC = 'BUSAN' WHERE DEPTNO = 20;
03 DELETE FROM DEPT_TCL WHERE DEPTNO = 40;
04 SELECT * FROM DEPT_TCL;
```

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO
50	NETWORK	SEOUL

실습 11-5 COMMIT으로 명령어 반영하기

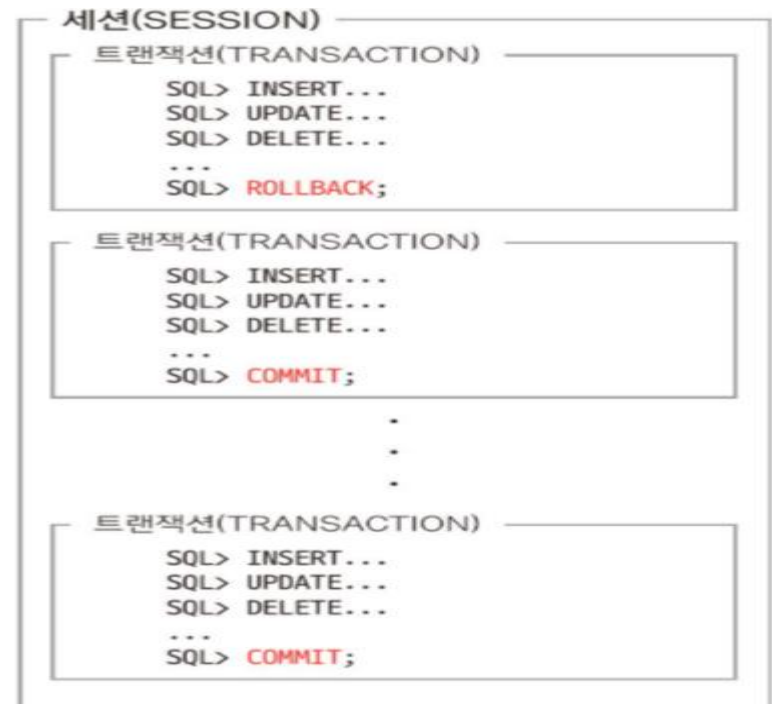
```
01 COMMIT;
```

11-3 세션과 읽기 일관성의 의미

▶ 세션이란?

- ▶ 데이터베이스 접속을 시작으로 접속을 종료하기까지 전체 기간
- ▶ 하나의 세션에는 여러 개의 트랜잭션이 존재

```
Microsoft Windows [Version 10.0.17134.285]  
(c) 2018 Microsoft Corporation. All rights reserved.  
  
C:\Users\weasyspublishing>sqlplus scott/tiger  
  
SQL*Plus: Release 11.2.0.1.0 Production on 금 9월 28 23:33:29 2018  
Copyright (c) 1982, 2018, Oracle. All rights reserved.  
  
다음에 접속됨:  
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Pr  
With the Partitioning, OLAP, Data Mining and Real Application Testin  
SQL>
```



하나의 세션 안에는 여러 개의 트랜잭션이 존재합니다.

11-3 세션과 읽기 일관성의 의미

▶ 읽기 일관성

- ▶ 특정 세션에서 수행하는 데이터의 변경이 확정되기 전까지, 다른 세션에서는 본래의 데이터를 보여줌

실습 11-6 토드와 SQL*PLUS로 세션 알아보기

세션 A(토드)

SELECT * FROM DEPT_TCL; -①

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO
50	NETWORK	SEOUL

세션 B(SQL * PLUS)

SELECT * FROM DEPT_TCL; -②

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO
50	NETWORK	SEOUL

11-3 세션과 읽기 일관성의 의미

▶ 읽기 일관성

- ▶ 특정 세션에서 수행하는 데이터의 변경이 확정되기 전까지, 다른 세션에서는 본래의 데이터를 보여줌

실습 11-7 토드와 SQL*PLUS로 세션 알아보기

세션 A(토드)	세션 B(SQL * PLUS)
DELETE FROM DEPT_TCL WHERE DEPTNO = 50; -①	세션 A의 DELETE 명령이 끝날 때까지 기다려 주세요.
SELECT * FROM DEPT_TCL; -②	SELECT * FROM DEPT_TCL; -③

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO

:: 결과 화면

SQL> SELECT * FROM DEPT_TCL;		
DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO
50	NETWORK	SEOUL
SQL>		

11-3 세션과 읽기 일관성의 의미

▶ 읽기 일관성

- ▶ 특정 세션에서 수행하는 데이터의 변경이 확정되기 전까지, 다른 세션에서는 본래의 데이터를 보여줌

실습 11-8 토드와 SQL*PLUS로 세션 알아보기

세션 A(토드)	세션 B(SQL * PLUS)
COMMIT; — ❶	세션 A의 COMMIT 명령이 끝날 때까지 기다려 주세요.
SELECT * FROM DEPT_TCL; — ❷	SELECT * FROM DEPT_TCL; — ❸

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO

:: 결과 화면

SQL> SELECT * FROM DEPT_TCL;		
DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO
SQL>		

11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK이란?

- ▶ 데이터 잠금
- ▶ 조작 중인 데이터를 다른 세션이 조작할 수 없음

실습 11-9 토드와 SQL*PLUS로 LOCK 알아보기

세션 A(토드)

SELECT * FROM DEPT_TCL; —1

:: 결과 화면

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO

세션 B(SQL * PLUS)

SELECT * FROM DEPT_TCL; —2

:: 결과 화면

SQL> SELECT * FROM DEPT_TCL;		
DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	SALES	CHICAGO
SQL>		

11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK이란?

- ▶ 데이터 잠금
- ▶ 조작 중인 데이터를 다른 세션이 조작할 수 없음

실습 11-10 토드와 SQL*PLUS로 LOCK 알아보기

세션 A(토드)	세션 B(SQL * PLUS)																								
UPDATE DEPT_TCL SET LOC='SEOUL ' WHERE DEPTNO = 30; — ①	세션 A의 UPDATE 명령이 끝날 때까지 기다려 주세요.																								
SELECT * FROM DEPT_TCL; — ②	SELECT * FROM DEPT_TCL; — ③																								
<p>:: 결과 화면</p> <table><tr><th>DEPTNO</th><th>DNAME</th><th>LOC</th></tr><tr><td>10</td><td>ACCOUNTING</td><td>NEW YORK</td></tr><tr><td>20</td><td>RESEARCH</td><td>BUSAN</td></tr><tr><td>30</td><td>SALES</td><td>SEOUL</td></tr></table>	DEPTNO	DNAME	LOC	10	ACCOUNTING	NEW YORK	20	RESEARCH	BUSAN	30	SALES	SEOUL	<p>:: 결과 화면</p> <div>SQL> SELECT * FROM DEPT_TCL;</div> <table><tr><th>DEPTNO</th><th>DNAME</th><th>LOC</th></tr><tr><td>10</td><td>ACCOUNTING</td><td>NEW YORK</td></tr><tr><td>20</td><td>RESEARCH</td><td>BUSAN</td></tr><tr><td>30</td><td>SALES</td><td>CHICAGO</td></tr></table> <div>SQL></div>	DEPTNO	DNAME	LOC	10	ACCOUNTING	NEW YORK	20	RESEARCH	BUSAN	30	SALES	CHICAGO
DEPTNO	DNAME	LOC																							
10	ACCOUNTING	NEW YORK																							
20	RESEARCH	BUSAN																							
30	SALES	SEOUL																							
DEPTNO	DNAME	LOC																							
10	ACCOUNTING	NEW YORK																							
20	RESEARCH	BUSAN																							
30	SALES	CHICAGO																							

11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK이란?

- ▶ 데이터 잠금
- ▶ 조작 중인 데이터를 다른 세션이 조작할 수 없음

실습 11-11 토드와 SQL*PLUS로 LOCK 알아보기

세션 A(토드)	세션 B(SQL * PLUS)
A는 아무런 작업을 하지 않습니다.	UPDATE DEPT_TCL SET DNAME='DATABASE' WHERE DEPTNO = 30; —①

```
SQL> UPDATE DEPT_TCL SET DNAME='DATABASE'  
2 WHERE DEPTNO = 30;
```

실습 11-11 이후의 SQL*PLUS 화면

실습 11-12 토드와 SQL*PLUS로 LOCK 알아보기

세션 A(토드)	세션 B(SQL * PLUS)
COMMIT; —①	세션 A의 COMMIT 명령어가 실행되는 순간의 변화를 확인합니다.

:: 결과 화면(SQL*PLUS)

```
SQL> UPDATE DEPT_TCL SET DNAME='DATABASE'  
2 WHERE DEPTNO = 30;  
1 행이 갱신되었습니다.  
SQL>
```

11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK이란?

- ▶ 데이터 잠금
- ▶ 조작 중인 데이터를 다른 세션이 조작할 수 없음

실습 11-13 토드와 SQL*PLUS로 LOCK 알아보기

세션 A(토드)	세션 B(SQL * PLUS)
SELECT * FROM DEPT_TCL;	SELECT * FROM DEPT_TCL;

실습 11-14 토드와 SQL*PLUS로 LOCK 알아보기 - 세션 B(SQL*PLUS)

```
01 COMMIT;
```

실습 11-15 토드와 SQL*PLUS로 LOCK 알아보기 - 세션 A(토드)

```
01 SELECT * FROM DEPT_TCL;
```

:: 결과 화면(실습 11-15)

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	BUSAN
30	DATABASE	SEOUL

11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK 종류

▶ 행 레벨 록

- ▶ 조작중인 행에 LOCK

▶ 테이블 레벨 록

- ▶ DML를 사용하여 데이터가 변경중인 테이블에 LOCK
- ▶ 데이터정의어(DDL) 사용 불가



11-4 수정 중인 데이터 접근을 막는 LOCK

- ▶ LOCK 종류
 - ▶ 행 레벨 록
 - ▶ 조작중인 행에 LOCK

```
UPDATE DEPT_TCL SET LOC = 'SEOUL';
```

```
DELETE FROM DEPT_TCL;
```

11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK 종류

▶ 테이블 레벨 록

- ▶ DML를 사용하여 데이터가 변경 중인 테이블에 LOCK
- ▶ 데이터정의어(DDL) 사용 불가

하지만 테이블에 변경되는 행의 수와는 상관없이 데이터 조작 명령어를 사용하여 데이터가 변경 중인 테이블은 테이블 단위 잠금이라는 의미로 '테이블 레벨 록(table level lock)'이 걸리게 됩니다. 즉 데이터를 변경 중인 세션 외 다른 세션에서 12장에서 살펴볼 데이터 정의어(DDL)를 통한 테이블의 구조를 변경할 수는 없습니다.

데이터 조작 관련 SQL문을 어떤 방식으로 작성하느냐에 따라 테이블의 일부 데이터만 LOCK이 될 수도 있고 테이블 전체 데이터가 LOCK이 될 수도 있다는 점을 기억하세요.



11-4 수정 중인 데이터 접근을 막는 LOCK

▶ LOCK 종류

▶ 테이블 레벨 록

- ▶ DML를 사용하여 데이터가 변경 중인 테이블에 LOCK
- ▶ 데이터정의어(DDL) 사용 불가

