

21장. 저장프로시저 함수 커서

4. 부서번호를 전달하여 해당 부서의 이름과 사원수, 최고 급여, 평균 급여를 구해보세요.(FOR~IN 사용합니다.)

```
1  CREATE OR REPLACE PROCEDURE CURSOR_TEST(
2      VDEPTNO IN EMP.DEPTNO%TYPE
3  ) IS
4      VEMPCOUNT NUMBER;
5  BEGIN
6      SELECT COUNT(*) INTO VEMPCOUNT FROM EMP WHERE DEPTNO = VDEPTNO;
7
8      IF VEMPCOUNT = 0 THEN
9          DBMS_OUTPUT.PUT_LINE('There is no employees in Dept ' || VDEPTNO || '.');
10     ELSE
11         FOR VEMP IN (SELECT DNAME
12                        , MAX(SAL) MAXSAL
13                        , ROUND(AVG(SAL), 2) AVGSAL
14                        FROM EMP
15                        NATURAL JOIN DEPT
16                        WHERE DEPTNO = VDEPTNO
17                        GROUP BY DEPTNO, DNAME)
18         LOOP
19             DBMS_OUTPUT.PUT_LINE('DNAME : ' || VEMP.DNAME);
20             DBMS_OUTPUT.PUT_LINE('EMP_COUNT : ' || VEMPCOUNT);
21             DBMS_OUTPUT.PUT_LINE('MAX_SAL : ' || VEMP.MAXSAL);
22             DBMS_OUTPUT.PUT_LINE('AVG_SAL : ' || VEMP.AVGSAL);
23         END LOOP;
24
25         /*
26          LOOP문 안에서 한 행씩 처리될 때마다 VEMPCOUNT := VEMPCOUNT + 1을 하고,
27          이 부분에서 IF문을 사용해, 해당 값이 0인 경우 '해당 부서의 사원이 존재하지 않습니다.'라고 처리하는 방법도 있다.
28          */
29     END IF;
30 END;
31 /
32
33 /*****
34 SCOTT > @CH21_QUIZ_04
35
36 Procedure created.
37
38 SCOTT > exec cursor_test(10)
39 DNAME : ACCOUNTING
40 EMP_COUNT : 3
41 MAX_SAL : 5000
42 AVG_SAL : 2916.67
43
44 PL/SQL procedure successfully completed.
45
46 SCOTT > exec cursor_test(50)
47 There is no employees in Dept 50.
48
49 PL/SQL procedure successfully completed.
50 *****/
```

22장. 패키지과 트리거

1. <탄탄히 다지기> DEPT01 테이블에 UPDATE가 되면 자동으로 TEMP에 이전 데이터를 저장하는 트리거

```

1 CREATE OR REPLACE TRIGGER DEPT01_UPDATE
2     AFTER UPDATE
3     ON DEPT01
4     FOR EACH ROW
5 BEGIN
6     INSERT INTO TEMP VALUES (:OLD.DEPTNO, :OLD.DNAME, :OLD.LOC);
7 END;
8 /

```

2. <탄탄히 다지기> 성적 트리거 작성하기

- GRADECARD 테이블에 데이터를 입력하면 TOT, AVG, GRADE를 자동으로 구하는 트리거를 작성해 보세요.
- INSERT INTO GRADECARD(NO, NAME, KOR, ENG, MATH) VALUES (GRADECARD_SEQ.NEXTVAL, 'TEST1', 90, 90, 90);
- TEST > SELECT * FROM GRADECARD;

NO NAME KOR ENG MAT TOT AVG GRADE

```

1  -- CREATE SEQUENCE ( GRADECARD_SEQ.sql )
2  CREATE SEQUENCE GRADECARD_SEQ
3      START WITH 1
4      INCREMENT BY 1;
5
6  -- CREATE FUNCTION ( CAL_GRADE_FUNC.sql )
7  CREATE OR REPLACE FUNCTION CAL_GRADE(
8      VSCORE IN GRADECARD.AVG%TYPE)
9      RETURN VARCHAR2
10     IS
11     VGRADE VARCHAR2(4);
12 BEGIN
13     SELECT GRADE
14     INTO VGRADE
15     FROM GRADE
16     WHERE VSCORE BETWEEN LOWSCORE AND HISCORE;
17
18     RETURN VGRADE;
19 END;
20 /
21
22 -- CREATE TRIGGER ( TRIG_GRADE.sql )
23 CREATE OR REPLACE TRIGGER TRIG_GRADE
24     BEFORE INSERT
25     ON GRADECARD
26     FOR EACH ROW
27 BEGIN
28     :NEW.TOT := :NEW.KOR + :NEW.ENG + :NEW.MAT;
29     :NEW.AVG := ROUND(:NEW.TOT / 3);
30     SELECT GRADE
31     INTO :NEW.GRADE
32     FROM GRADE
33     WHERE :NEW.AVG BETWEEN LOWSCORE AND HISCORE;
34     -- 📌 아래처럼 할 필요가 없음, :NEW.TOT처럼 위에서 작성했기 때문에 INSERT되면서 자동으로 데이터가 삽입된다.
35     -- :NEW.GRADE := CAL_GRADE(:NEW.AVG);
36     -- UPDATE GRADECARD
37     -- SET TOT = :NEW.TOT,
38     --     AVG = :NEW.AVG,
39     --     GRADE = :NEW.GRADE
40     -- WHERE NO = :NEW.NO;
41 END;
42 /
43
44 -- SEQUENCE, FUNCTION, TRIGGER sql파일 실행하기

```

```

45 SCOTT > @GRADECARD_SEQ
46 Sequence created.
47
48 SCOTT > @CAL_GRADE_FUNC
49 Function created.
50
51 SCOTT > @TRIG_GRADE
52 Trigger created.
53
54 -- GRADECARD에 insert문 수행
55 INSERT INTO GRADECARD(NO, NAME, KOR, ENG, MAT) VALUES (GRADECARD_SEQ.NEXTVAL, 'TEST1', 90, 90, 90);
56 INSERT INTO GRADECARD(NO, NAME, KOR, ENG, MAT) VALUES (GRADECARD_SEQ.NEXTVAL, 'TEST9', 33, 76, 48);

```

NO	NAME	KOR	ENG	MAT	TOT	AVG	GRADE
21	TEST1	90	90	90	270	90	A0
22	TEST1	97	76	30	203	68	D+
1	TEST1	90	90	90	270	90	A0

게시판 쿼리

1. 사용자 board/1234 생성합니다.

```

1 -- conn system/oracle
2 CREATE USER board IDENTIFIED BY 1234;
3 -- grant resource, connect to board;
4 -- conn board/1234

```

2. 테이블 생성

```

1 CREATE TABLE BOARD
2 (
3     BOARD_NUM        NUMBER(5),      -- 글번호(기본키)
4     BOARD_NAME        VARCHAR2(30),  -- 작성자
5     BOARD_PASS        VARCHAR2(30),  -- 비밀번호
6     BOARD_SUBJECT     VARCHAR2(300),  -- 제목
7     BOARD_CONTENT     VARCHAR2(4000), -- 내용
8     BOARD_FILE        VARCHAR2(50),  -- 첨부될 파일 명(시스템에 저장되는 파일명)
9     BOARD_ORIGINAL    VARCHAR2(50),  -- 첨부될 파일 명(시스템에 저장된 파일의 원래 파일명)
10    BOARD_RE_REF       NUMBER(5),     -- 답변 글 작성시 참조되는 글의 번호
11    BOARD_RE_LEV       NUMBER(5),     -- 답변 글의 깊이(원문 글:0, 답변:1, 답변의답변:2)
12    BOARD_RE_SEQ       NUMBER(5),     -- 답변 글의 순서(원문 글 기준으로 보여주는 순서)
13    BOARD_READCOUNT  NUMBER(5),     -- 글의 조회수
14    BOARD_DATE         DATE,          -- 글의작성날짜
15    PRIMARY KEY (BOARD_NUM)
16 );

```

3. 글 번호는 시퀀스를 사용합니다.

```

1 -- 시퀀스 이름 : board_seq
2 -- 시작값 : 1
3 -- 증가값 : 1
4 CREATE SEQUENCE BOARD_SEQ
5     START WITH 1
6     INCREMENT BY 1;

```

4. 테이블 조회와 같이 원문글이 저장되도록 INSERT문을 작성하세요.

```
1  -- 조건1) 원문글의 경우 BOARD_RE_REF 컬럼의 값은 원문 글 번호와 같습니다.
2  -- 조건2) BOARD_RE_LEV,BOARD_RE_SEQ,BOARD_READCOUNT 값은 모두 0 입니다.
3  -- 조건3) BOARD_DATE는 시스템 날짜입니다.
4  -- 조건4) BOARD_FILE, BOARD_ORIGINAL는 null로 처리합니다.
5  -- 조건5) 비밀번호 삽입은 하지만 조회 시 제외합니다.
6  INSERT INTO BOARD
7  VALUES
8  (
9      BOARD_SEQ.NEXTVAL -- BOARD_NUM
10     , 'admin' -- BOARD_NAME
11     , 'admin_pass' -- BOARD_PASS
12     , 'JSP 공부는 재미있어요' -- BOARD_SUBJECT
13     , '열심히 해봐요' -- BOARD_CONTENT
14     , NULL -- BOARD_FILE
15     , NULL -- BOARD_ORIGINAL
16     , BOARD_SEQ.NEXTVAL -- BOARD_RE_REF
17     , 0 -- BOARD_RE_LEV
18     , 0 -- BOARD_RE_SEQ
19     , 0 -- BOARD_READCOUNT
20     , SYSDATE -- BOARD_DATE
21 );
22
23 /*
24  ✨ SEQUENCE의 NEXTVAL은 하나의 문장당 한번만 수행되어 위와 같이 작성해도 동일한 값이 들어간다.
25  */
```

BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DATE
1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06

5. 원문글에 답변을 달도록 쿼리문을 작성하세요.(INSERT문)

```
1  -- 조건1) 글 번호는 시퀀스로 증가합니다.
2  -- 조건2) BOARD_RE_REF 컬럼의 값은 원문글의 글 번호와 일치합니다.
3  -- 조건3) BOARD_RE_LEV,BOARD_RE_SEQ는 원문글의 BOARD_RE_LEV,BOARD_RE_SEQ 보다 1씩 증가합니다.
4  -- 조건4) BOARD_DATE는 시스템 날짜입니다.
5  INSERT INTO BOARD
6  SELECT BOARD_SEQ.NEXTVAL,
7      'admin',
8      'admin_pass',
9      'JAVA도',
10     'JAVA도',
11     NULL,
12     NULL,
13     BOARD_RE_REF,
14     BOARD_RE_LEV + 1,
15     BOARD_RE_SEQ + 1,
16     0,
17     SYSDATE
18 FROM BOARD
19 WHERE BOARD_NUM = 1;
20
21 /*
22  ✨ 원문글 클릭 후, 답변을 달때 원문의 _REF, _LEV, _SEQ의 값은 꼭 같이 넘겨줘야한다. ( 본 과제의 경우, 실제 웹에서의 프로세스를 가정하지 않고, 게시판 쿼리의 전반적인 과정을 이해하기위한 목적이 크므로 위와 같이 쿼리 작성함 )
23
24  ✨ INSERT INTO BOARD
25     VALUES ( BOARD_SEQ.NEXTVAL, 'admin', 'admin_pass', 'JAVA도', 'JAVA도', null, null, 1, 1, 1, 0, SYSDATE );
26  */
```

BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DATE
1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
2	admin	JAVA도	JAVA도	null	null	1	1	1	0	2024-08-27 02:27:39

6. 테이블 조회는 **BOARD_RE_REF**를 기준으로 내림차순으로 정렬하고, **BOARD_RE_REF** 같으면 **BOARD_RE_SEQ**를 오름차순 기준으로 정렬합니다. 정렬기준으로 처음부터 10개를 조회합니다.(SELECT문 사용)

```
1 SELECT *
2 FROM (SELECT ROWNUM AS RNUM, E.*
3       FROM (SELECT BOARD_NUM
4             , BOARD_NAME
5             , BOARD_SUBJECT
6             , BOARD_CONTENT
7             , BOARD_FILE
8             , BOARD_ORIGINAL
9             , BOARD_RE_REF
10            , BOARD_RE_LEV
11            , BOARD_RE_SEQ
12            , BOARD_READCOUNT
13            , BOARD_DATE
14           FROM BOARD
15           ORDER BY BOARD_RE_REF DESC, BOARD_RE_SEQ) E
16      WHERE ROWNUM <= 5)
17 WHERE RNUM BETWEEN 1 AND 5;
```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DATE
1	1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
2	2	admin	JAVA도	JAVA도	null	null	1	1	1	0	2024-08-27 02:27:39

7. 원문글에 답변을 추가하는 경우 기존 글들의 **BOARD_RE_SEQ**는 1증가합니다. (UPDATE문 사용)

```
1 -- 조건) BOARD_RE_REF가 원문글의 번호이고
2 -- BOARD_RE_SEQ값이 원문글의 BOARD_RE_SEQ보다 큰 경우에만 답변의 BOARD_RE_SEQ는 1증가합니다
3 UPDATE BOARD
4 SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
5 WHERE BOARD_RE_REF = (SELECT BOARD_RE_REF
6                       FROM BOARD
7                       WHERE BOARD_NUM = 1)
8    AND BOARD_RE_SEQ > (SELECT BOARD_RE_SEQ
9                       FROM BOARD
10                      WHERE BOARD_NUM = 1);
11
12 /*
13 ✨ 트랜잭션으로 처리되어야할 로직임. 실제 환경에서는 위 처럼 서버쿼리를 사용할 필요가 없다. 답변의 부모가 되는 원문의 데이터를 가져올 수 있기
14 때문에, 그것을 새로운 답변 내용들과 함께 서버로 전달해버리면 됨 (쉽게 생각할 것!)
15
16 UPDATE BOARD
17     SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
18     WHERE BOARD_RE_REF = 1
19     AND BOARD_RE_SEQ > 0;
20 */
```

BOARD_NUM	BOARD_SUBJECT	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ
1	JSP 공부는 재미있어요	1	0	0
2	JAVA도	1	1	2

8. 원문글(**BOARD_NUM=1**)에 답변을 추가로 달도록 쿼리문을 작성합니다.(INSERT문)
(최신의 답변글이 원문글 바로 아래 위치합니다.)

```
1  INSERT INTO BOARD
2  SELECT BOARD_SEQ.NEXTVAL,
3         'admin',
4         'admin_pass',
5         '나도 찬성',
6         '찬성입니다.',
7         NULL,
8         NULL,
9         BOARD_RE_REF,
10        BOARD_RE_LEV + 1,
11        BOARD_RE_SEQ + 1,
12        0,
13        SYSDATE
14 FROM BOARD
15 WHERE BOARD_NUM = 1;
16
17 /*
18  ✨ INSERT INTO BOARD
19    VALUES ( BOARD_SEQ.NEXTVAL, 'admin', 'admin_pass', '나도 찬성', '찬성입니다.', null, null, 1, 1, 1, 0, SYSDATE
20  );
21  */
```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	1	admin	JSP 공부는 재미있어 요	열심히 해박요	null	null	1	0	0	0	2024-08-27 02:04:06
2	3	admin	나도 찬성	찬성입니다.	null	null	1	1	1	0	2024-08-27 03:35:34
3	2	admin	JAVA도	JAVA도	null	null	1	1	2	0	2024-08-27 02:27:39

9. 글 번호 2(BOARD_NUM=2)에 답변을 합니다.(INSERT 문)

```
1  -- 조건1) BOARD_RE_REF는 원문글번호를 가집니다.
2  -- 조건2) BOARD_RE_LEV는 글번호 2의 BOARD_RE_LEV 보다 1 증가합니다.
3  -- 조건3) BOARD_RE_SEQ는 글번호 2의 BOARD_RE_SEQ 보다 1증가합니다.
4  INSERT INTO BOARD
5  SELECT BOARD_SEQ.NEXTVAL,
6         'admin',
7         'admin_pass',
8         '나도 찬성이요',
9         '찬성',
10        NULL,
11        NULL,
12        BOARD_RE_REF,
13        BOARD_RE_LEV + 1,
14        BOARD_RE_SEQ + 1,
15        0,
16        SYSDATE
17 FROM BOARD
18 WHERE BOARD_NUM = 2;
19
20 /*
21  ✨ INSERT INTO BOARD
22    VALUES ( BOARD_SEQ.NEXTVAL, 'admin', 'admin_pass', '나도 찬성이요', '찬성', null, null, 1, 2, 3, 0, SYSDATE );
23  */
```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	1	admin	JSP 공부는 재미있어 요	열심히 해박요	null	null	1	0	0	0	2024-08-27 02:04:06
2	3	admin	나도 찬성	찬성입니다.	null	null	1	1	1	0	2024-08-27 03:35:34
3	2	admin	JAVA도	JAVA도	null	null	1	1	2	0	2024-08-27 02:27:39
4	4	admin	나도 찬성이요	찬성	null	null	1	2	3	0	2024-08-27 03:40:53

10. 원문글(BOARD_NUM=1)에 답변을 합니다.다음 조건에 맞는 쿼리문을 작성하세요(UPDATE)

```

1  -- 조건1) 글 번호 3,2,4의 BOARD_RE_SEQ는 기존의 값에서 1씩 증가합니다.
2  -- BOARD_RE_REF의 값이 원문글 번호이고
3  -- BOARD_RE_SEQ값이 원문글의 BOARD_RE_SEQ보다 큰 경우에만 답변의 BOARD_RE_SEQ는 1증가합니다.
4  UPDATE BOARD
5  SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
6  WHERE BOARD_RE_REF = (SELECT BOARD_RE_REF
7                        FROM BOARD
8                        WHERE BOARD_NUM = 1)
9  AND BOARD_RE_SEQ > (SELECT BOARD_RE_SEQ
10                     FROM BOARD
11                     WHERE BOARD_NUM = 1);
12
13 /*
14  ✨
15  UPDATE BOARD
16  SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
17  WHERE BOARD_RE_REF = 1
18  AND BOARD_RE_SEQ > 0;
19 */

```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	1	admin	JSP 공부는 재미있어 요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
2	3	admin	나도 찬성	찬성입니다.	null	null	1	1	2	0	2024-08-27 03:35:34
3	2	admin	JAVA도	JAVA도	null	null	1	1	3	0	2024-08-27 02:27:39
4	4	admin	나도 찬성이요	찬성	null	null	1	2	4	0	2024-08-27 03:40:53

```

1  -- 조건2) 새로 입력하는 답변글의 BOARD_RE_REF는 원문글 번호 입니다. (INSERT)
2  -- BOARD_RE_LEV는 원문글의 BOARD_RE_LEV보다 1증가 합니다.
3  -- BOARD_RE_SEQ 는 원문글의 BOARD_RE_SEQ 보다 1증가 합니다.
4  INSERT INTO BOARD
5  SELECT BOARD_SEQ.NEXTVAL,
6  'admin',
7  'admin_pass',
8  '저도요',
9  '저도요',
10 NULL,
11 NULL,
12 BOARD_RE_REF,
13 BOARD_RE_LEV + 1,
14 BOARD_RE_SEQ + 1,
15 0,
16 SYSDATE
17 FROM BOARD
18 WHERE BOARD_NUM = 1;
19
20 /*
21  ✨ INSERT INTO BOARD
22  VALUES ( BOARD_SEQ.NEXTVAL, 'admin', 'admin_pass', '저도요', '저도요', null, null, 1, 1, 1, 0, SYSDATE );
23 */

```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	1	admin	JSP 공부는 재미있어 요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
2	5	admin	저도요	저도요	null	null	1	1	1	0	2024-08-27 03:45:11
3	3	admin	나도 찬성	찬성입니다.	null	null	1	1	2	0	2024-08-27 03:35:34
4	2	admin	JAVA도	JAVA도	null	null	1	1	3	0	2024-08-27 02:27:39
5	4	admin	나도 찬성이요	찬성	null	null	1	2	4	0	2024-08-27 03:40:53

11. 원문 글 삽입

```

1  INSERT INTO BOARD
2  VALUES
3  (
4      BOARD_SEQ.NEXTVAL -- BOARD_NUM
5      , 'admin' -- BOARD_NAME
6      , 'admin_pass' -- BOARD_PASS
7      , '힘내세요' -- BOARD_SUBJECT
8      , '힘내세요' -- BOARD_CONTENT
9      , NULL -- BOARD_FILE
10     , NULL -- BOARD_ORIGINAL
11     , BOARD_SEQ.CURRVAL -- BOARD_RE_REF
12     , 0 -- BOARD_RE_LEV
13     , 0 -- BOARD_RE_SEQ
14     , 0 -- BOARD_READCOUNT
15     , SYSDATE -- BOARD_DATE
16 );

```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	6	admin	힘내세요	힘내세요	null	null	6	0	0	0	2024-08-27 03:46:21
2	1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
3	5	admin	저도요	저도요	null	null	1	1	1	0	2024-08-27 03:45:11
4	3	admin	나도 찬성	찬성입니다.	null	null	1	1	2	0	2024-08-27 03:35:34
5	2	admin	JAVA도	JAVA도	null	null	1	1	3	0	2024-08-27 02:27:39
6	4	admin	나도 찬성이요	찬성	null	null	1	2	4	0	2024-08-27 03:40:53

12. 6번글에 답변을 달아요.

```

1  -- 6번글에 대해 존재하던 답변이 없어서 아래 쿼리 수행해도 변화 없음
2  -- 기존재하는 답변이 있었다면, update문의 where절에 의해서 BOARD_RE_SEQ가 update됐을것.
3  UPDATE BOARD
4  SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
5  WHERE BOARD_RE_REF = (SELECT BOARD_RE_REF
6                        FROM BOARD
7                        WHERE BOARD_NUM = 6)
8  AND BOARD_RE_SEQ > (SELECT BOARD_RE_SEQ
9                    FROM BOARD
10                   WHERE BOARD_NUM = 6);
11
12  /*
13  UPDATE BOARD
14  SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
15  WHERE BOARD_RE_REF = 6
16  AND BOARD_RE_SEQ > 0;
17  */
18
19  INSERT INTO BOARD
20  SELECT BOARD_SEQ.NEXTVAL,
21         'admin',
22         'admin_pass',
23         'Re: 힘내세요',
24         'Re: 힘내세요',
25         NULL,
26         NULL,
27         BOARD_RE_REF,
28         BOARD_RE_LEV + 1,
29         BOARD_RE_SEQ + 1,

```



```

30         0,
31         SYSDATE
32 FROM BOARD
33 WHERE BOARD_NUM = 6;
34 /*
35     ✨
36 INSERT INTO BOARD
37     VALUES ( BOARD_SEQ.NEXTVAL, 'admin', 'admin_pass', 'Re: 힘내세요', 'Re: 힘내세요', null, null, 6, 1, 1, 0,
38     SYSDATE );
39 */

```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	6	admin	힘내세요	힘내세요	null	null	6	0	0	0	2024-08-27 03:46:21
2	7	admin	Re: 힘내세요	Re: 힘내세요	null	null	6	1	1	0	2024-08-27 03:48:40
3	1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
4	5	admin	저도요	저도요	null	null	1	1	1	0	2024-08-27 03:45:11
5	3	admin	나도 찬성	찬성입니다.	null	null	1	1	2	0	2024-08-27 03:35:34
6	2	admin	JAVA도	JAVA도	null	null	1	1	3	0	2024-08-27 02:27:39
7	4	admin	나도 찬성이요	찬성	null	null	1	2	4	0	2024-08-27 03:40:53

13. 3번 글에 대한 답변을 합니다.

```

1  -- 조건1) 2, 4번 글에 대한 BOARD_RE_SEQ를 기존 값에서 1증가 합니다. (8번 참조)
2  -- 조건2) 8번 글의 BOARD_RE_REF는 1, BOARD_RE_LEV은 답변의 답변이므로 2, BOARD_RE_SEQ는 3번 글의 답변이므로 3번 글의
3  -- BOARD_RE_SEQ보다 1증가 합니다.
4  -- 즉, 답변은 답변을 다는 글의 BOARD_RE_LEV, BOARD_RE_SEQ 보다 각각 1씩 증가합니다.
5  UPDATE BOARD
6  SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
7  WHERE BOARD_RE_REF = (SELECT BOARD_RE_REF
8                        FROM BOARD
9                        WHERE BOARD_NUM = 3)
10 AND BOARD_RE_SEQ > (SELECT BOARD_RE_SEQ
11                     FROM BOARD
12                     WHERE BOARD_NUM = 3);
13 /*
14     ✨
15 UPDATE BOARD
16     SET BOARD_RE_SEQ = BOARD_RE_SEQ + 1
17     WHERE BOARD_RE_REF = 1
18     AND BOARD_RE_SEQ > 2;
19 */

```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	6	admin	힘내세요	힘내세요	null	null	6	0	0	0	2024-08-27 03:46:21
2	7	admin	Re: 힘내세요	Re: 힘내세요	null	null	6	1	1	0	2024-08-27 03:48:40
3	1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
4	5	admin	저도요	저도요	null	null	1	1	1	0	2024-08-27 03:45:11
5	3	admin	나도 찬성	찬성입니다.	null	null	1	1	2	0	2024-08-27 03:35:34
6	2	admin	JAVA도	JAVA도	null	null	1	1	4	0	2024-08-27 02:27:39
7	4	admin	나도 찬성이요	찬성	null	null	1	2	5	0	2024-08-27 03:40:53

```

1  -- 조건2) 8번 글의 BOARD_RE_REF는 1, BOARD_RE_LEV은 답변의 답변이므로 2,
2  -- BOARD_RE_SEQ는 3번 글의 답변이므로 3번 글의 BOARD_RE_SEQ보다 1증가 합니다.
3  -- 즉, 답변은 답변을 다는 글의 BOARD_RE_LEV, BOARD_RE_SEQ 보다 각각 1씩 증가합니다.

```

```
4  INSERT INTO BOARD
5  SELECT BOARD_SEQ.NEXTVAL,
6         'admin',
7         'admin_pass',
8         'RE: 나도 찬성',
9         '하하',
10        NULL,
11        NULL,
12        BOARD_RE_REF,
13        BOARD_RE_LEV + 1,
14        BOARD_RE_SEQ + 1,
15        0,
16        SYSDATE
17  FROM BOARD
18  WHERE BOARD_NUM = 3;
19
20  /*
21   *
22   * INSERT INTO BOARD
23   *   VALUES ( BOARD_SEQ.NEXTVAL, 'admin', 'admin_pass', 'RE: 나도 찬성', '하하', null, null, 1, 2, 3, 0, SYSDATE
24   * );
25   */
```

RNUM	BOARD_NUM	BOARD_NAME	BOARD_SUBJECT	BOARD_CONTENT	BOARD_FILE	BOARD_ORIGINAL	BOARD_RE_REF	BOARD_RE_LEV	BOARD_RE_SEQ	BOARD_READCOUNT	BOARD_DAT
1	6	admin	힘내세요	힘내세요	null	null	6	0	0	0	2024-08-27 03:46:21
2	7	admin	Re: 힘내세요	Re: 힘내세요	null	null	6	1	1	0	2024-08-27 03:48:40
3	1	admin	JSP 공부는 재미있어요	열심히 해봐요	null	null	1	0	0	0	2024-08-27 02:04:06
4	5	admin	저도요	저도요	null	null	1	1	1	0	2024-08-27 03:45:11
5	3	admin	나도 찬성	찬성입니다.	null	null	1	1	2	0	2024-08-27 03:35:34
6	8	admin	RE: 나도 찬성	하하	null	null	1	2	3	0	2024-08-27 05:11:07
7	2	admin	JAVA도	JAVA도	null	null	1	1	4	0	2024-08-27 02:27:39
8	4	admin	나도 찬성이요	찬성	null	null	1	2	5	0	2024-08-27 03:40:53

삭제의 조건)

- 조건1) 해당 글이 원문글인 경우 이 원문글의 **BOARD_RE_REF**와 같은 글들을 삭제합니다.
- 조건2) 해당 글이 원문글이 아닌 경우 자신과 자신의 하위 레벨 답변글들도 삭제합니다.

14. 글 번호 2번을 삭제하면 4번 글이 삭제되도록 합니다. (글 번호 3번을 삭제하면 8번글도 삭제됩니다.)

```
1  | // TO-DO
```

15. 글 번호 4번을 삭제하면 4번 글만 삭제됩니다.

```
1  | // TO-DO
```

16. 글 번호 1번을 삭제하면 2, 3, 4, 5, 8번 글이 삭제됩니다.

```
1  | // TO-DO
```