

6-3. 로우를 컬럼으로

홍형경

chariehong@gmail.com

2020.01

1. 테이블에 있는 로우를 컬럼 형태로 변환하기

- score_table 생성

```
CREATE TABLE score_table (  
    YEARS    VARCHAR2(4), -- 연도  
    GUBUN    VARCHAR2(30), -- 구분(중간/기말)  
    SUBJECTS VARCHAR2(30), -- 과목  
    SCORE    NUMBER );    -- 점수
```

1. 로우를 컬럼으로

· score_table 데이터 입력

```
INSERT INTO score_table VALUES('2019','중간고사','국어',92);  
INSERT INTO score_table VALUES('2019','중간고사','영어',87);  
INSERT INTO score_table VALUES('2019','중간고사','수학',67);  
INSERT INTO score_table VALUES('2019','중간고사','과학',80);  
INSERT INTO score_table VALUES('2019','중간고사','지리',93);  
INSERT INTO score_table VALUES('2019','중간고사','독일어',82);  
INSERT INTO score_table VALUES('2019','기말고사','국어',88);  
INSERT INTO score_table VALUES('2019','기말고사','영어',80);  
INSERT INTO score_table VALUES('2019','기말고사','수학',93);  
INSERT INTO score_table VALUES('2019','기말고사','과학',91);  
INSERT INTO score_table VALUES('2019','기말고사','지리',89);  
INSERT INTO score_table VALUES('2019','기말고사','독일어',83);  
COMMIT;
```

1. 로우를 컬럼으로

· score_table 조회

Select *

from score_table;

	YEARS	GUBUN	SUBJECTS	SCORE
1	2019	중간고사	국어	92
2	2019	중간고사	영어	87
3	2019	중간고사	수학	67
4	2019	중간고사	과학	80
5	2019	중간고사	지리	93
6	2019	중간고사	독일어	82
7	2019	기말고사	국어	88
8	2019	기말고사	영어	80
9	2019	기말고사	수학	93
10	2019	기말고사	과학	91
11	2019	기말고사	지리	89
12	2019	기말고사	독일어	83

1. 로우를 컬럼으로

```
SELECT years,  
       gubun,  
       CASE WHEN subjects = '국어' THEN score ELSE 0 END "국어",  
       CASE WHEN subjects = '영어' THEN score ELSE 0 END "영어",  
       CASE WHEN subjects = '수학' THEN score ELSE 0 END "수학",  
       CASE WHEN subjects = '과학' THEN score ELSE 0 END "과학",  
       CASE WHEN subjects = '지리' THEN score ELSE 0 END "지리",  
       CASE WHEN subjects = '독일어' THEN score ELSE 0 END "독일어"  
FROM score_table a;
```

	YEARS	GUBUN	국어	영어	수학	과학	지리	독일어
1	2019	중간고사	92	0	0	0	0	0
2	2019	중간고사	0	87	0	0	0	0
3	2019	중간고사	0	0	67	0	0	0
4	2019	중간고사	0	0	0	80	0	0
5	2019	중간고사	0	0	0	0	93	0
6	2019	중간고사	0	0	0	0	0	82
7	2019	기말고사	88	0	0	0	0	0
8	2019	기말고사	0	80	0	0	0	0
9	2019	기말고사	0	0	93	0	0	0
10	2019	기말고사	0	0	0	91	0	0
11	2019	기말고사	0	0	0	0	89	0
12	2019	기말고사	0	0	0	0	0	83

1. 로우를 컬럼으로

```
SELECT years, gubun,  
       SUM(국어) AS 국어, SUM(영어) AS 영어, SUM(수학) AS 수학,  
       SUM(과학) AS 과학, SUM(지리) AS 지리, SUM(독일어) AS 독일어  
FROM (  
  SELECT years, gubun,  
         CASE WHEN subjects = '국어' THEN score ELSE 0 END "국어",  
         CASE WHEN subjects = '영어' THEN score ELSE 0 END "영어",  
         CASE WHEN subjects = '수학' THEN score ELSE 0 END "수학",  
         CASE WHEN subjects = '과학' THEN score ELSE 0 END "과학",  
         CASE WHEN subjects = '지리' THEN score ELSE 0 END "지리",  
         CASE WHEN subjects = '독일어' THEN score ELSE 0 END "독일어"  
  FROM score_table a  
)  
GROUP BY years, gubun;
```

	YEARS	GUBUN	국어	영어	수학	과학	지리	독일어
1	2019	중간고사	92	87	67	80	93	82
2	2019	기말고사	88	80	93	91	89	83

1. 로우를 컬럼으로 – decode로 변환

```
SELECT years, gubun,  
       SUM(국어) AS 국어, SUM(영어) AS 영어, SUM(수학) AS 수학,  
       SUM(과학) AS 과학, SUM(지리) AS 지리, SUM(독일어) AS 독일어  
FROM (  
  SELECT years, gubun,  
         DECODE(subjects,'국어',score,0) "국어",  
         DECODE(subjects,'영어',score,0) "영어",  
         DECODE(subjects,'수학',score,0) "수학",  
         DECODE(subjects,'과학',score,0) "과학",  
         DECODE(subjects,'지리',score,0) "지리",  
         DECODE(subjects,'독일어',score,0) "독일어"  
  FROM score_table a  
)  
GROUP BY years, gubun;
```

	YEARS	GUBUN	국어	영어	수학	과학	지리	독일어
1	2019	중간고사	92	87	67	80	93	82
2	2019	기말고사	88	80	93	91	89	83

1. 로우를 컬럼으로 – WITH 절 사용

```
WITH mains AS ( SELECT years, gubun,  
    CASE WHEN subjects = '국어' THEN score ELSE 0 END "국어",  
    CASE WHEN subjects = '영어' THEN score ELSE 0 END "영어",  
    CASE WHEN subjects = '수학' THEN score ELSE 0 END "수학",  
    CASE WHEN subjects = '과학' THEN score ELSE 0 END "과학",  
    CASE WHEN subjects = '지리' THEN score ELSE 0 END "지리",  
    CASE WHEN subjects = '독일어' THEN score ELSE 0 END "독일어"  
FROM score_table a  
)  
SELECT years, gubun,  
    SUM(국어) AS 국어, SUM(영어) AS 영어, SUM(수학) AS 수학,  
    SUM(과학) AS 과학, SUM(지리) AS 지리, SUM(독일어) AS 독일어  
FROM mains  
GROUP BY years, gubun;
```

	YEARS	GUBUN	국어	영어	수학	과학	지리	독일어
1	2019	중간고사	92	87	67	80	93	82
2	2019	기말고사	88	80	93	91	89	83

1. 로우를 컬럼으로 – PIVOT 절 사용

```
SELECT *  
FROM ( SELECT years, gubun, subjects, score  
        FROM score_table )  
PIVOT ( SUM(score)  
        FOR subjects IN ( '국어', '영어', '수학', '과학', '지리', '독일어')  
        );
```

	YEARS	GUBUN	국어	영어	수학	과학	지리	독일어
1	2019	중간고사	92	87	67	80	93	82
2	2019	기말고사	88	80	93	91	89	83

2. 컬럼을 로우로

· score_col_table 생성

```
CREATE TABLE score_col_table (  
    YEARS    VARCHAR2(4), -- 연도  
    GUBUN    VARCHAR2(30), -- 구분(중간/기말)  
    KOREAN    NUMBER,      -- 국어점수  
    ENGLISH  NUMBER,      -- 영어점수  
    MATH     NUMBER,      -- 수학점수  
    SCIENCE  NUMBER,      -- 과학점수  
    GEOLOGY  NUMBER,      -- 지리점수  
    GERMAN   NUMBER,      -- 독일어점수  
);
```

2. 컬럼을 로우로

- score_col_table 데이터 입력

```
INSERT INTO score_col_table  
VALUES ('2019', '중간고사', 92, 87, 67, 80, 93, 82 );
```

```
INSERT INTO score_col_table  
VALUES ('2019', '기말고사', 88, 80, 93, 91, 89, 83 );
```

```
COMMIT;
```

2. 컬럼을 로우로 – UNION ALL 사용

```
SELECT YEARS, GUBUN, '국어' AS SUBJECT, KOREAN AS SCORE
```

```
FROM score_col_table
```

```
UNION ALL
```

```
SELECT YEARS, GUBUN, '영어' AS SUBJECT, ENGLISH AS SCORE
```

```
FROM score_col_table
```

```
UNION ALL
```

```
SELECT YEARS, GUBUN, '수학' AS SUBJECT, MATH AS SCORE
```

```
FROM score_col_table
```

```
UNION ALL
```

```
SELECT YEARS, GUBUN, '과학' AS SUBJECT, SCIENCE AS SCORE
```

```
FROM score_col_table
```

```
UNION ALL
```

```
SELECT YEARS, GUBUN, '지리' AS SUBJECT, GEOLOGY AS SCORE
```

```
FROM score_col_table
```

```
UNION ALL
```

```
SELECT YEARS, GUBUN, '독일어' AS SUBJECT, GERMAN AS SCORE
```

```
FROM score_col_table
```

```
ORDER BY 1, 2 DESC;
```

	YEARS	GUBUN	SUBJECT	SCORE
1	2019	중간고사	독일어	82
2	2019	중간고사	수학	67
3	2019	중간고사	지리	93
4	2019	중간고사	국어	92
5	2019	중간고사	과학	80
6	2019	중간고사	영어	87
7	2019	기말고사	지리	89
8	2019	기말고사	과학	91
9	2019	기말고사	독일어	83
10	2019	기말고사	영어	80
11	2019	기말고사	국어	88
12	2019	기말고사	수학	93

2. 컬럼을 로우로 – UNPIVOT절 사용

```
SELECT *  
  FROM score_col_table  
UNPIVOT ( score  
          FOR subjects IN ( KOREAN  AS '국어',  
                           ENGLISH AS '영어',  
                           MATH     AS '수학',  
                           SCIENCE AS '과학',  
                           GEOLOGY AS '지리',  
                           GERMAN  AS '독일어'  
          )  
);
```

	YEARS	GU...	SUBJECTS	SCORE
1	2019	중간고사	국어	92
2	2019	중간고사	영어	87
3	2019	중간고사	수학	67
4	2019	중간고사	과학	80
5	2019	중간고사	지리	93
6	2019	중간고사	독일어	82
7	2019	기말고사	국어	88
8	2019	기말고사	영어	80
9	2019	기말고사	수학	93
10	2019	기말고사	과학	91
11	2019	기말고사	지리	89
12	2019	기말고사	독일어	83

2. 컬럼을 로우로 – PL/SQL – Table Function 사용

```
CREATE OR REPLACE TYPE obj_subject AS OBJECT (  
    YEARS    VARCHAR2(4), -- 연도  
    GUBUN    VARCHAR2(30), -- 구분(중간/기말)  
    SUBJECTS VARCHAR2(30), -- 과목  
    SCORE    NUMBER      -- 점수  
);
```

```
CREATE OR REPLACE TYPE subject_nt IS TABLE OF obj_subject;
```

2. 컬럼을 로우로 – PL/SQL – Table Function 사용

```
CREATE OR REPLACE FUNCTION fn_pipe_table_ex
```

```
RETURN subject_nt
```

```
PIPELINED
```

```
IS
```

```
vp_cur SYS_REFCURSOR;
```

```
v_cur score_col_table%ROWTYPE;
```

```
-- 반환할 컬렉션 변수 선언 (컬렉션 타입이므로 초기화를 한다)
```

```
vnt_return subject_nt := subject_nt();
```

```
BEGIN
```

```
-- SYS_REFCURSOR 변수로 ch14_score_col_table 테이블을 선택해 커서를 오픈
```

```
OPEN vp_cur FOR SELECT * FROM score_col_table ;
```

```
-- 루프를 돌며 입력 매개변수 vp_cur를 v_cur로 패치
```

```
LOOP
```

```
FETCH vp_cur INTO v_cur;
```

```
EXIT WHEN vp_cur%NOTFOUND;
```

```
-- 컬렉션 타입이므로 EXTEND 메소드를 사용해 한 로우씩 신규 삽입
```

```
vnt_return.EXTEND();
```

```
-- 컬렉션 변수에 커서 변수의 값 할당
```

```
vnt_return(vnt_return.LAST).YEARS := v_cur.YEARS;
```

```
vnt_return(vnt_return.LAST).GUBUN := v_cur.GUBUN;
```

```
vnt_return(vnt_return.LAST).SUBJECTS := '국어';
```

```
vnt_return(vnt_return.LAST).SCORE := v_cur.KOREAN;
```

```
PIPE ROW ( vnt_return(vnt_return.LAST)); -- 국어 반환
```

```
vnt_return(vnt_return.LAST).SUBJECTS := '영어';
```

```
vnt_return(vnt_return.LAST).SCORE := v_cur.ENGLISH;
```

```
PIPE ROW ( vnt_return(vnt_return.LAST)); -- 영어 반환
```

```
....
```

```
.....
```

```
END LOOP;
```

```
RETURN;
```

```
END;
```

2. 컬럼을 로우로 – PL/SQL – Table Function 사용

```
SELECT *  
FROM TABLE ( fn_pipe_table_ex );
```

	YEARS	GUBUN	SUBJECTS	SCORE
1	2019	중간고사	국어	92
2	2019	중간고사	영어	87
3	2019	중간고사	수학	67
4	2019	중간고사	과학	80
5	2019	중간고사	지리	93
6	2019	중간고사	독일어	82
7	2019	기말고사	국어	88
8	2019	기말고사	영어	80
9	2019	기말고사	수학	93
10	2019	기말고사	과학	91
11	2019	기말고사	지리	89
12	2019	기말고사	독일어	83

3. 정리

- 로우를 컬럼 형태로 변환해 조회하는 경우가 많음
- 어떤 결과를 얻기 위해 SQL을 작성하는 방법은 매우 다양함