

NEW OPEN SQL

17682004 김동환

OPEN SQL은 SAP에 사용되는 모든 데이터베이스에서 사용할 수 있도록 정의된 SQL 구문이다.

NEW OPEN SQL 은 HANA 데이터베이스에서만 사용되는 것이 아니라 SQP에 설치된 모든 데이터베이스에서 동작한다. SAP NETWEAVER 7.4 이후부터 NEW OPEN SQL을 사용할 수 있다.

NEW OPEN SQL의 특징

CODE PUSHDOWN (Calculations into DB) 방식

기존 OPEN SQL에서 사용하던 기능들을 지원한다. (ex. FOR ALL ENTRIES, Aggregate 함수)

변수 앞에 @를 사용한다.

컬럼 구분을 위해 쉼표 (,)기호를 사용한다.

사칙연산, CASE 조건식, 문자열 연결, 형 변환 SELECT문에서 처리 가능하다.

Colaesce 사용해서 NULL 대체 가능하다.

Constant Value 사용 가능하다.

JOIN 구문의 확장이 가능하다.

CODE PUSHDOWN (Calculations into DB)

NEW OPEN SQL과 OPEN SQL의 차이점은 데이터를 Calculation하는 시점이다.

기존 OPEN SQL은 AS ABAP=> DB => CALCULATION AS ABAP으로 Application Layer에서 Calculation이 이루어지고 NEW OPEN SQL은 AS ABAP => CALCULATION DB => AS ABAP으로 Database Layer에서 Calculation이 이루어 지게 된다.

사용 예제

@기호와 쉼표(,) 기호 사용

```
□ TYPES: BEGIN OF wa,
    fldate LIKE sbook-fldate,
    count TYPE i,
    avg TYPE p DECIMALS 2,
END OF wa.

DATA: gt_itab TYPE TABLE OF wa,
      gs_itab like LINE OF gt_itab.

SELECT fldate, COUNT( * ), AVG( luggweight )
FROM sbook
INTO TABLE @gt_itab
WHERE carrid = 'LH'
AND connid = '0400'
GROUP BY fldate.
```

NEW OPEN SQL에서는 변수 앞에 @를 사용하고 컬럼을 나누기 위해 쉼표 (,)를 사용한다.

COUNT와 AVG 같은 기존의 Aggregate 함수 사용이 가능하다.

DIV, MOD, ABS, 계산식, DIRECT 변수사용

```
DATA(offset) = 10000.

SELECT id, num1, num2,
       div( num1, num2 ) AS div,
       mod( num1, num2 ) AS mod,
       @offset + abs( num1 - num2 ) AS sum
FROM demo_expressions

INTO TABLE @data(results).

cl_demo_output=>display( results ).
```

RESULTS

ID	NUM1	NUM2	DIV	MOD	SUM
0	18	18	1	0	10000
1	51	9	5	6	10042
2	46	77	0	46	10031
3	33	54	0	33	10021
4	91	95	0	91	10004
5	21	87	0	21	10066
6	38	44	0	38	10006
7	11	94	0	11	10083
8	13	35	0	13	10022
9	11	93	0	11	10082

NEW OPEN SQL 문의 SELCET에서 DIV(몫), MOD(나머지), ABS(절대값) 계산식을 사용하여 DIRECT 변수 선언한 results에 데이터를 가져오도록 사용 가능하다.

CEIL, FLOOR, DIRECT 변수 사용

```
SELECT price, ceil( price ) AS ceil_value
FROM sflight
INTO TABLE @DATA(ceil).

cl_demo_output=>display( ceil ).

SELECT price, floor( price ) AS floor_value
FROM sflight
INTO TABLE @DATA(floor).

cl_demo_output=>display( floor ).
```

CEIL		FLOOR	
PRICE	CEIL_VALUE	PRICE	FLOOR_VALUE
422.94	423	422.94	422
422.94	423	422.94	422
1030.0	1030	1030.0	1030
422.94	423	422.94	422
422.94	423	422.94	422
422.94	423	422.94	422
422.94	423	422.94	422

SELECT 문에서 CEIL(올림)과 FLOOR(내림) 적용하고 DIRECT 변수에 데이터를 가져오도록 사용
CEIL과 FLOOR는 가장 가까운 정수로 올리거나 내림 한다.

문자열 연결

```
□ TYPES:BEGIN OF t_str,
|     idname TYPE c LENGTH 40,
|     END OF t_str.

DATA:gt_itab TYPE TABLE OF t_str,
      gs_itab LIKE LINE OF gt_itab.

SELECT carrid && '-' && carrname AS idname
FROM scarr
INTO CORRESPONDING FIELDS OF TABLE @gt_itab.

□ LOOP AT gt_itab INTO gs_itab.
|     WRITE:/ gs_itab-idname.
|     ENDLLOOP.
```

NEW OPEN SQL 내에서 CONCENTRATE 구문을 사용하지 않고 문자열을 && 기호를 사용하여 연결할 수 있다. 연결된 문자열의 최대 길이는 255자 이다.

CASE 구문 사용

```

TYPES:BEGIN OF t_str,
    carrid TYPE scarr-carrid,
    idname TYPE c LENGTH 40,
END OF t_str.

DATA:gt_itab TYPE TABLE OF t_str,
      gs_itab LIKE LINE OF gt_itab.

DATA else TYPE c LENGTH 10 VALUE 'Not available'.

SELECT carrid,
CASE carrid
    WHEN 'AC' THEN ( carrid && '-' && carrname )
    WHEN 'AZ' THEN ( 'Available' )
    ELSE @else
END AS idname
FROM scarr
INTO CORRESPONDING FIELDS OF TABLE @gt_itab.

```

GT_ITAB	
CARRID	IDNAME
AA	Not availa
AC	AC-Air Canada
AF	Not availa
AZ	Available
BA	Not availa
FJ	Not availa
CO	Not availa
DL	Not availa
AB	Not availa
LH	Not availa
NG	Not availa
JL	Not availa
NW	Not availa
QF	Not availa
SA	Not availa
SQ	Not availa
SR	Not availa
UA	Not availa
	Not availa
KO	Not availa
Air	Not availa

NEW OPEN SQL에서 CASE 구문을 이용하여 컬럼 값에 따라서 분기문으로 처리할 수 있다.

Coalesce 옵션 사용

```

TYPES:BEGIN OF t_str,
    carrid TYPE scarr-carrid,
    price TYPE i,
END OF t_str.

DATA:gt_itab TYPE TABLE OF t_str.

SELECT scarr~carrid, coalesce( price,1 ) AS price
FROM scarr LEFT OUTER JOIN sflight ON scarr~carrid = sflight~carrid
INTO CORRESPONDING FIELDS OF TABLE @gt_itab
WHERE scarr~carrid = 'KO'.

```

GT_ITAB	
CARRID	PRICE
KO	1

Coalesce 옵션은 괄호안의 컬럼의 값이 NULL일 경우 다른 값으로 대체하여 출력할 수 있다.

ABAP에서 데이터 선언 시 기본으로 Initial Value로 지정되고 NULL값과는 다른 값이다. 따라서 Coalesce 구문은 outer join과 같이 값이 존재하지 않는 경우에 활용 가능하다.

Constant Value 사용

```
DATA : gv_exist TYPE c.

PARAMETERS p_carrid TYPE scarr-carrid DEFAULT 'AA'.

SELECT SINGLE @abap_true
FROM scarr
WHERE carrid = @p_carrid
INTO @gv_exist.

IF gv_exist = abap_true.
    cl_demo_output=>display(
        |DATA exists in SCARR| ).
ELSE.
    cl_demo_output=>display(
        |DATA does not exist in SCARR| ).
ENDIF.
```

Output

DATA exists in SCARR

NEW OPEN SQL에서 SELECT LIST에 Constant value를 사용할 수 있으며 해당 record가 존재할 시 gv_exist에 X를 반환한다. 조건에 맞는 데이터 존재 여부확인시 활용이 가능하다.

JOIN

NEW OPEN SQL에서는 RIGHT JOIN이 추가적으로 지원된다. 또한 OUTER JOIN 시 WHERE 조건에 OUTER JOIN으로 사용된 컬럼이 조건에 포함될 수 있도록 변경되었다. JOIN 테이블과 Subquery의 최대 개수가 50개로 확장되었다.

```
TYPES : BEGIN OF t_str,
        carrid TYPE scarr-carrid, price TYPE i,
      END OF t_str.
DATA : gt_itab TYPE TABLE OF t_str.

SELECT a~carrid, price
FROM sflight AS a INNER JOIN scarr AS b ON a~carrid = b~carrid
AND a~carrid LIKE 'A%'
INTO CORRESPONDING FIELDS OF TABLE @gt_itab.
```

GT_ITAB	
CARRID	PRICE
AA	422
AA	422
AZ	1030
AA	422
AA	422

NEW OPEN SQL에서 JOIN의 ON 조건에 LIKE, BETWEEN, <, >과 기능을 사용 가능하다.

CROSS JOIN

```

DATA BEGIN OF wa.
  DATA mandt TYPE t000-mandt.
  DATA mtext TYPE t000-mtext.
  INCLUDE TYPE t100.
DATA END OF wa.
DATA itab LIKE STANDARD TABLE OF wa WITH EMPTY KEY.

SELECT t000~mandt, t000~mtext, t100~*
  FROM t000 CROSS JOIN t100
 WHERE t100~arbgb = 'SABAPDEMOS'
 ORDER BY t000~mandt, t100~sprsl, t100~msgnr
 INTO TABLE @itab.

cl_demo_output=>display( itab ).

```

T000 테이블

	MANDT	MTEXT
<input type="checkbox"/>	000	SAP AG
<input type="checkbox"/>	400	KON YANG

T100 테이블

	SPRSL	ARBGB	MSGNR	TEXT
<input type="checkbox"/>	3	SABAPDEMOS	001	A message ...
<input type="checkbox"/>	3	SABAPDEMOS	007	No entry found
<input type="checkbox"/>	3	SABAPDEMOS	050	Division by zero
<input type="checkbox"/>	3	SABAPDEMOS	777	Message with type &1 &2 in event &3
<input type="checkbox"/>	3	SABAPDEMOS	888	& & &
<input type="checkbox"/>	D	SABAPDEMOS	001	Eine Nachricht ...
<input type="checkbox"/>	D	SABAPDEMOS	007	Kein Eintrag gefunden
<input type="checkbox"/>	D	SABAPDEMOS	050	Division durch Null
<input type="checkbox"/>	D	SABAPDEMOS	777	Nachricht vom Typ &1 &2 zum Zeitpunkt &3.
<input type="checkbox"/>	D	SABAPDEMOS	888	& & &
<input type="checkbox"/>	E	SABAPDEMOS	001	A message ...
<input type="checkbox"/>	E	SABAPDEMOS	007	No entry found
<input type="checkbox"/>	E	SABAPDEMOS	050	Division by zero
<input type="checkbox"/>	E	SABAPDEMOS	777	Message with type &1 &2 in event &3
<input type="checkbox"/>	E	SABAPDEMOS	888	& & &

CROSS JOIN은 한쪽 테이블의 모든 행과 다른 테이블의 모든 행을 JOIN 시키는 기능을 한다.

Output					
ITAB					
	MANDT	MTEXT	SPRSL	ARBGB	MSGNR TEXT
000	SAP	AG	3	SABAPDEMOS	001 A message ...
000	SAP	AG	3	SABAPDEMOS	007 No entry found
000	SAP	AG	3	SABAPDEMOS	050 Division by zero
000	SAP	AG	3	SABAPDEMOS	777 Message with type &1 &2 in event &3
000	SAP	AG	3	SABAPDEMOS	888 & & &
000	SAP	AG	D	SABAPDEMOS	001 Eine Nachricht ...
000	SAP	AG	D	SABAPDEMOS	007 Kein Eintrag gefunden
000	SAP	AG	D	SABAPDEMOS	050 Division durch Null
000	SAP	AG	D	SABAPDEMOS	777 Nachricht vom Typ &1 &2 zum Zeitpunkt &3.
000	SAP	AG	D	SABAPDEMOS	888 & & &
000	SAP	AG	E	SABAPDEMOS	001 A message ...
000	SAP	AG	E	SABAPDEMOS	007 No entry found
000	SAP	AG	E	SABAPDEMOS	050 Division by zero
000	SAP	AG	E	SABAPDEMOS	777 Message with type &1 &2 in event &3
000	SAP	AG	E	SABAPDEMOS	888 & & &
400	KON	YANG	3	SABAPDEMOS	001 A message ...
400	KON	YANG	3	SABAPDEMOS	007 No entry found
400	KON	YANG	3	SABAPDEMOS	050 Division by zero
400	KON	YANG	3	SABAPDEMOS	777 Message with type &1 &2 in event &3
400	KON	YANG	3	SABAPDEMOS	888 & & &
400	KON	YANG	D	SABAPDEMOS	001 Eine Nachricht ...
400	KON	YANG	D	SABAPDEMOS	007 Kein Eintrag gefunden
400	KON	YANG	D	SABAPDEMOS	050 Division durch Null
400	KON	YANG	D	SABAPDEMOS	777 Nachricht vom Typ &1 &2 zum Zeitpunkt &3.
400	KON	YANG	D	SABAPDEMOS	888 & & &
400	KON	YANG	E	SABAPDEMOS	001 A message ...

참고 자료)

- NEW OPEN SQL (김성준 교수님)
- HELP.SAP