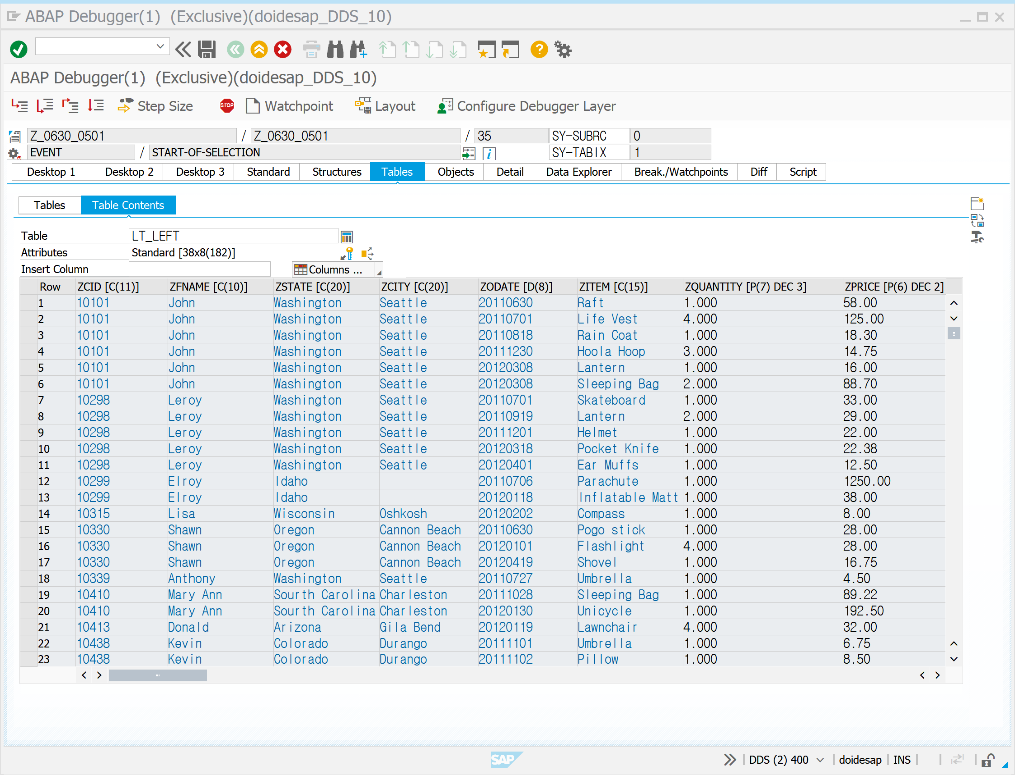
0630 | ABAP

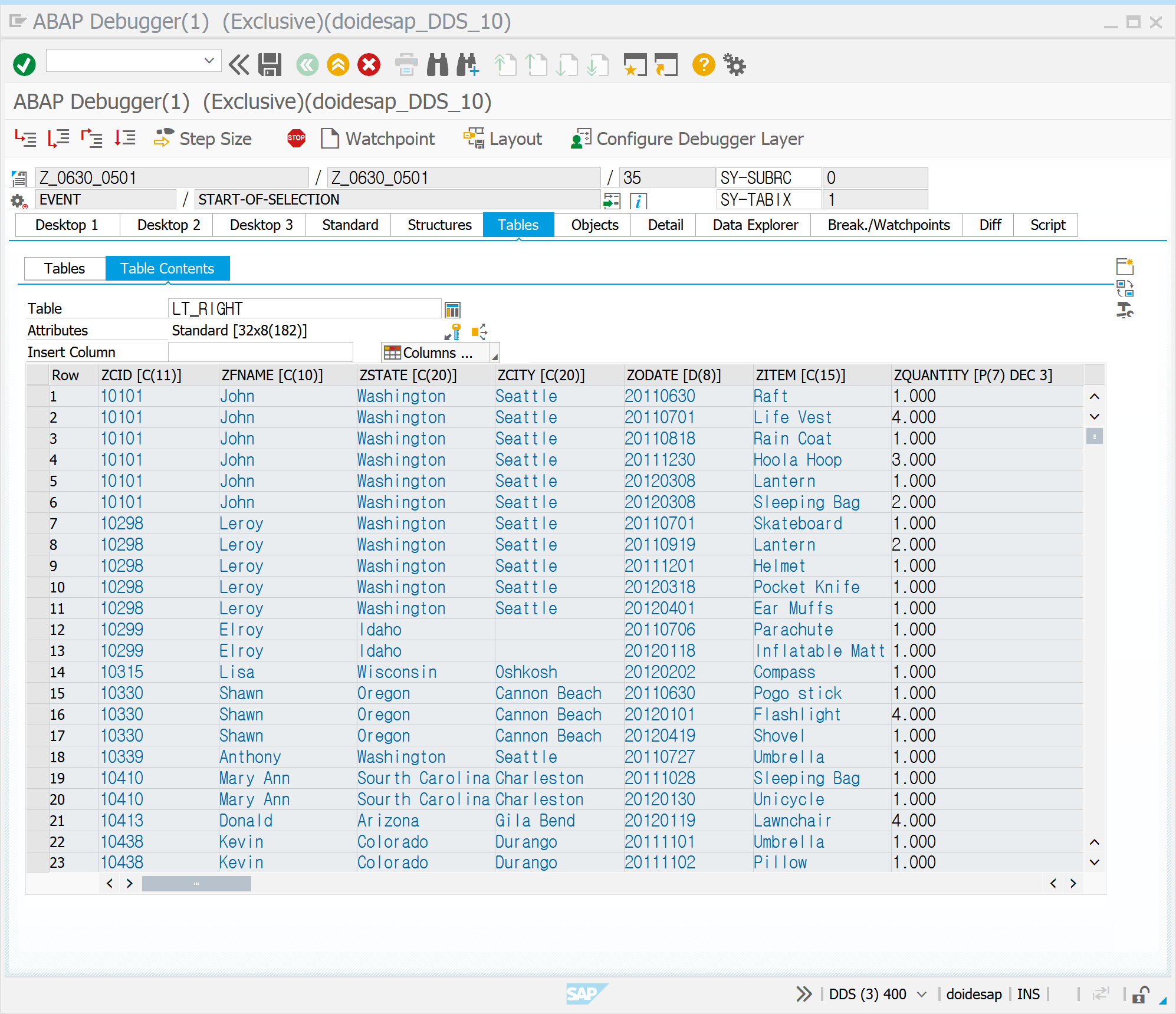
SE80 | 0501 | LEFT OUTER JOIN

DATA: BEGIN OF LS\_STR,  
  ZCID TYPE ZCUSTOMERS-ZCID,  
  ZFNAME TYPE ZCUSTOMERS-ZFNAME,  
  ZSTATE TYPE ZCUSTOMERS-ZSTATE,  
  ZCITY TYPE  ZCUSTOMERS-ZCITY,  
  ZODATE TYPE ZORDERS-ZODATE,  
  ZITEM TYPE ZORDERS-ZITEM,  
  ZQUANTITY TYPE ZORDERS-ZQUANTITY,  
  ZPRICE TYPE ZORDERS-ZPRICE,  
  END OF LS\_STR.  
  
DATA LT\_LEFT LIKE TABLE OF LS\_STR.  
DATA LT\_INNER LIKE TABLE OF LS\_STR.  
DATA LT\_RIGHT LIKE TABLE OF LS\_STR.  
  
\*LEFT OUTER JOIN JOIN  
SELECT C~ZCID, C~ZFNAME, C~ZSTATE, C~ZCITY,O~ZODATE,O~ZITEM,O~ZQUANTITY,O~ZPRICE  
  FROM ZCUSTOMERS AS C LEFT OUTER JOIN ZORDERS AS O  
  ON C~ZCID = O~ZCID  
  INTO TABLE @LT\_LEFT.



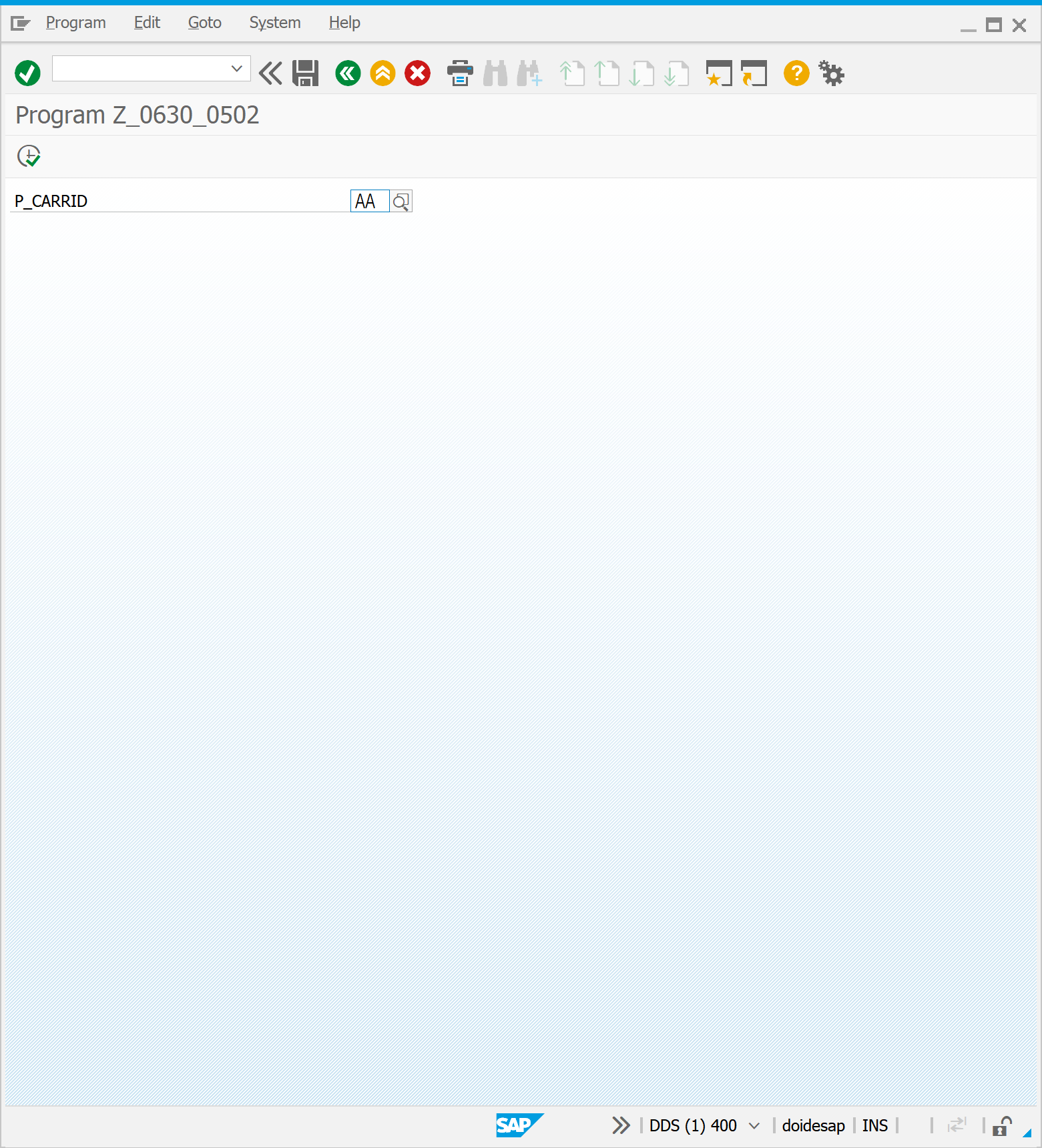
SE80 | 0501 | RIGHT OUTER JOIN

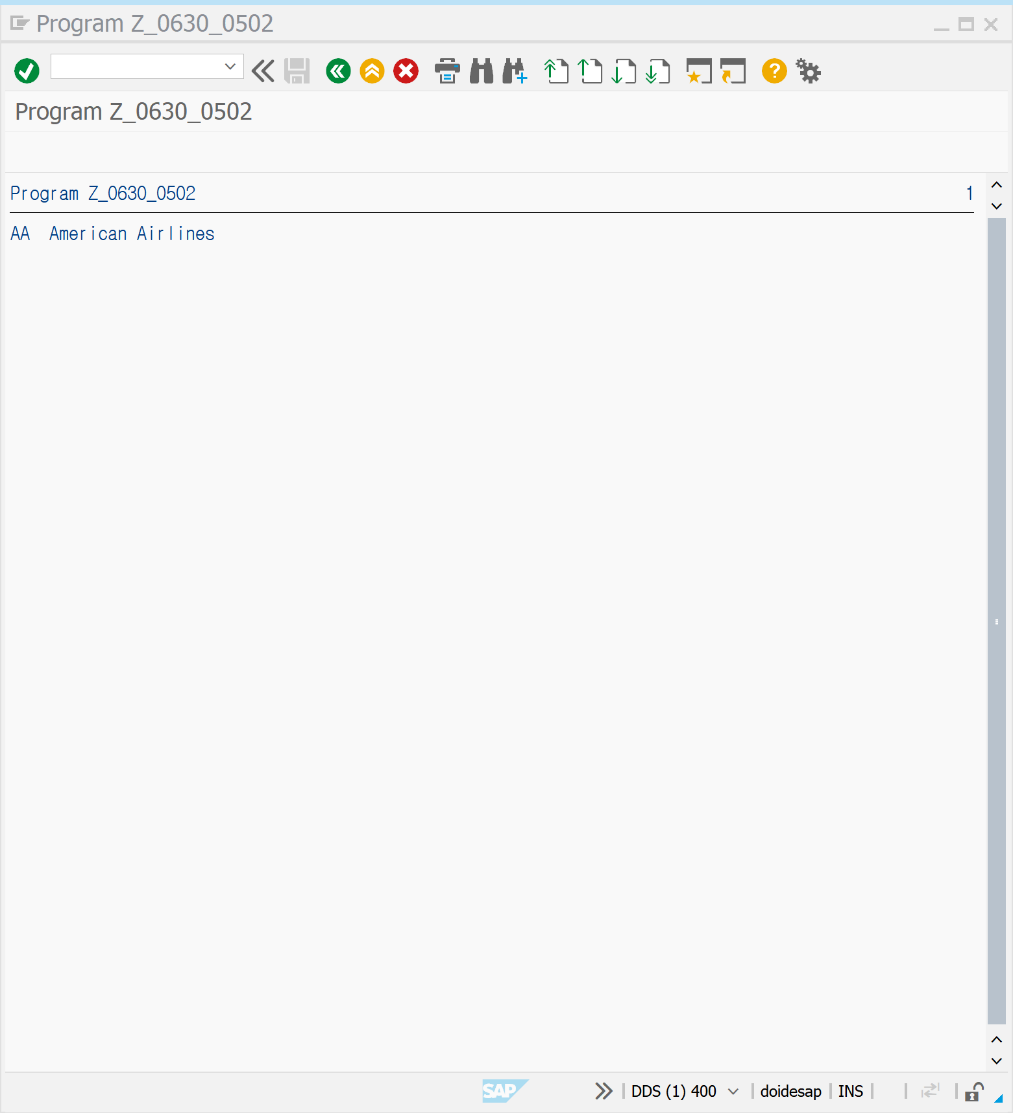
\*RIGHT OUTER JOIN     
  SELECT C~ZCID, C~ZFNAME, C~ZSTATE, C~ZCITY,O~ZODATE,O~ZITEM,O~ZQUANTITY,O~ZPRICE  
  FROM ZCUSTOMERS AS C RIGHT OUTER JOIN ZORDERS AS O  
  ON C~ZCID = O~ZCID  
  INTO TABLE @LT\_RIGHT.



SE80 | 0502 | NEW SQL QUERY | @ 사용하기

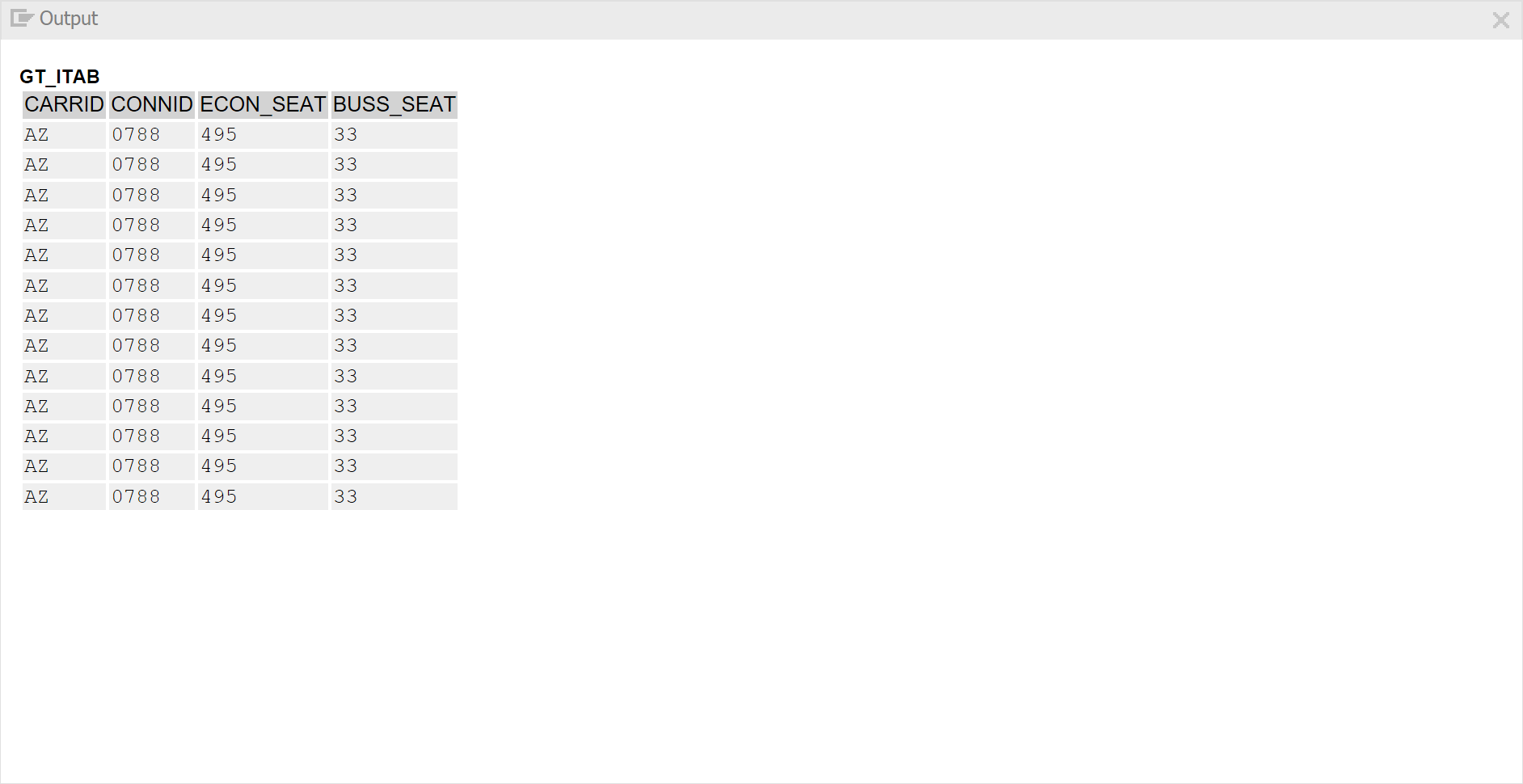
\*변수 앞에 @기호를 사용한다.  
\*변수 앞에 @를 사용하지 않는 CLASSICAL OPEN SQL은 폐기  
\*@는 필수사항이다  
  
PARAMETERS: P\_CARRID TYPE SCARR-CARRID.  
  
DATA: BEGIN OF GS\_DATA,  
           CARRID TYPE SCARR-CARRID,  
           CARRNAME TYPE SCARR-CARRNAME,  
            END OF GS\_DATA.  
  
\*컬럼 구분을 위해 , 기호를 사용한다.  
SELECT SINGLE CARRID, CARRNAME  
               INTO CORRESPONDING FIELDS OF @GS\_DATA  
               FROM SCARR  
               WHERE CARRID = @P\_CARRID.  
  
WRITE:/ GS\_DATA-CARRID, GS\_DATA-CARRNAME.





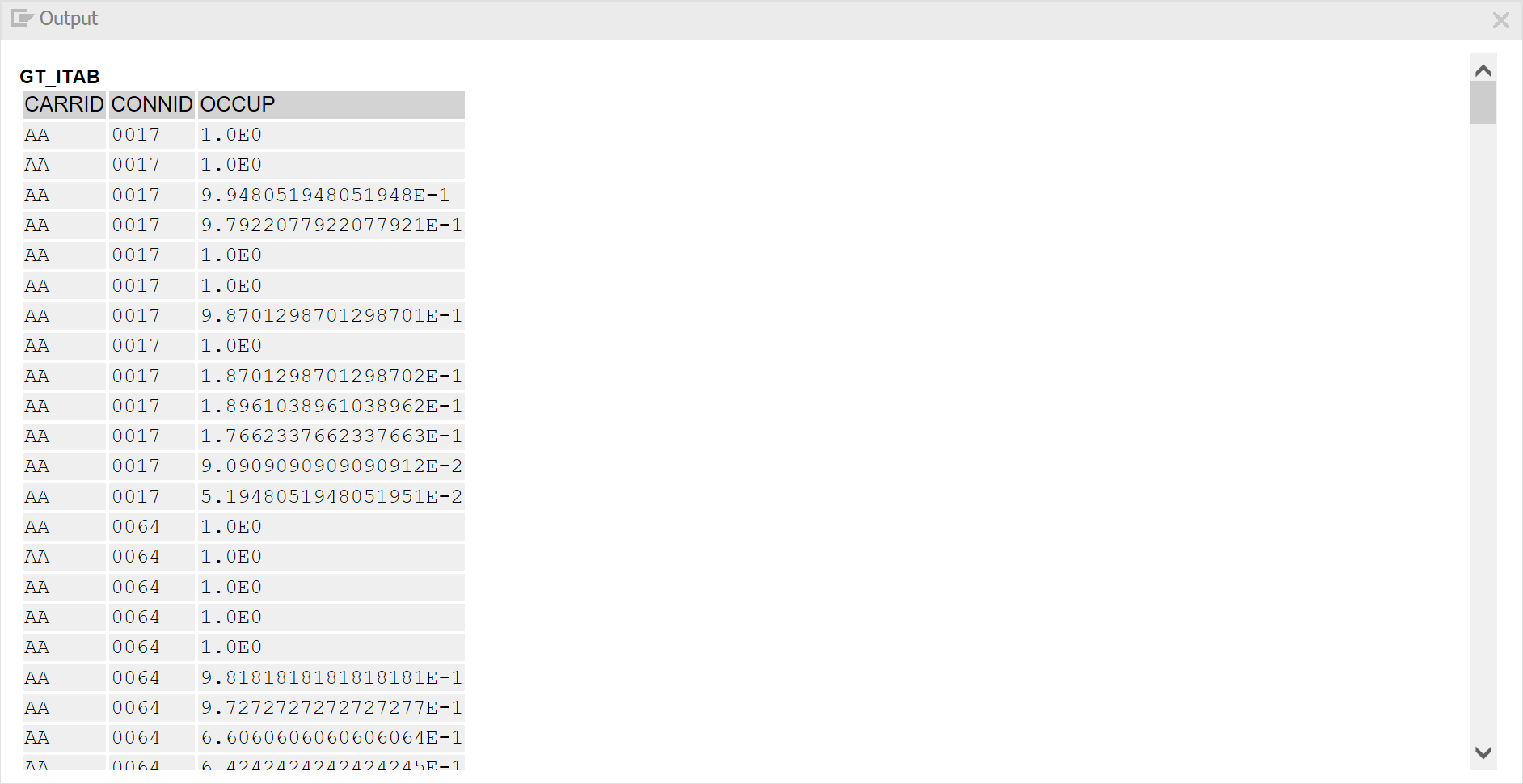
SE80 | 0503 | NEW SQL QUERY | 계산식 표현 사용 가능

\*계산식 표현이 사용 가능하다.  
\*계산식을 이용해서 다양한 값의 변형이 가능하다  
TYPES: BEGIN OF T\_SFLIGHT,  
             CARRID TYPE SFLIGHT-CARRID,  
             CONNID TYPE SFLIGHT-CONNID,  
             ECON\_SEAT TYPE SFLIGHT-SEATSMAX,  
             BUSS\_SEAT TYPE SFLIGHT-SEATSMAX\_B,  
             END OF T\_SFLIGHT.  
  
DATA: GT\_ITAB TYPE TABLE OF T\_SFLIGHT.  
DATA: GV\_SALE TYPE P VALUE '0.1' DECIMALS 1.  
DATA(GV\_ADD) = 20.  
  
\*AGGREGATE 함수의 기능 확장 AVG MAX MIN SUM COUNT 내에서 계산식이 가능하다  
SELECT CARRID, CONNID, SEATSMAX + @GV\_ADD AS ECON\_SEAT,  
              ( SEATSMAX\_B + ( SEATSMAX\_B \* @GV\_SALE ) ) AS BUSS\_SEAT  
               FROM SFLIGHT  
               INTO CORRESPONDING FIELDS OF TABLE @GT\_ITAB  
               WHERE CARRID = 'AZ' AND CONNID = '788'.  
  
CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ).



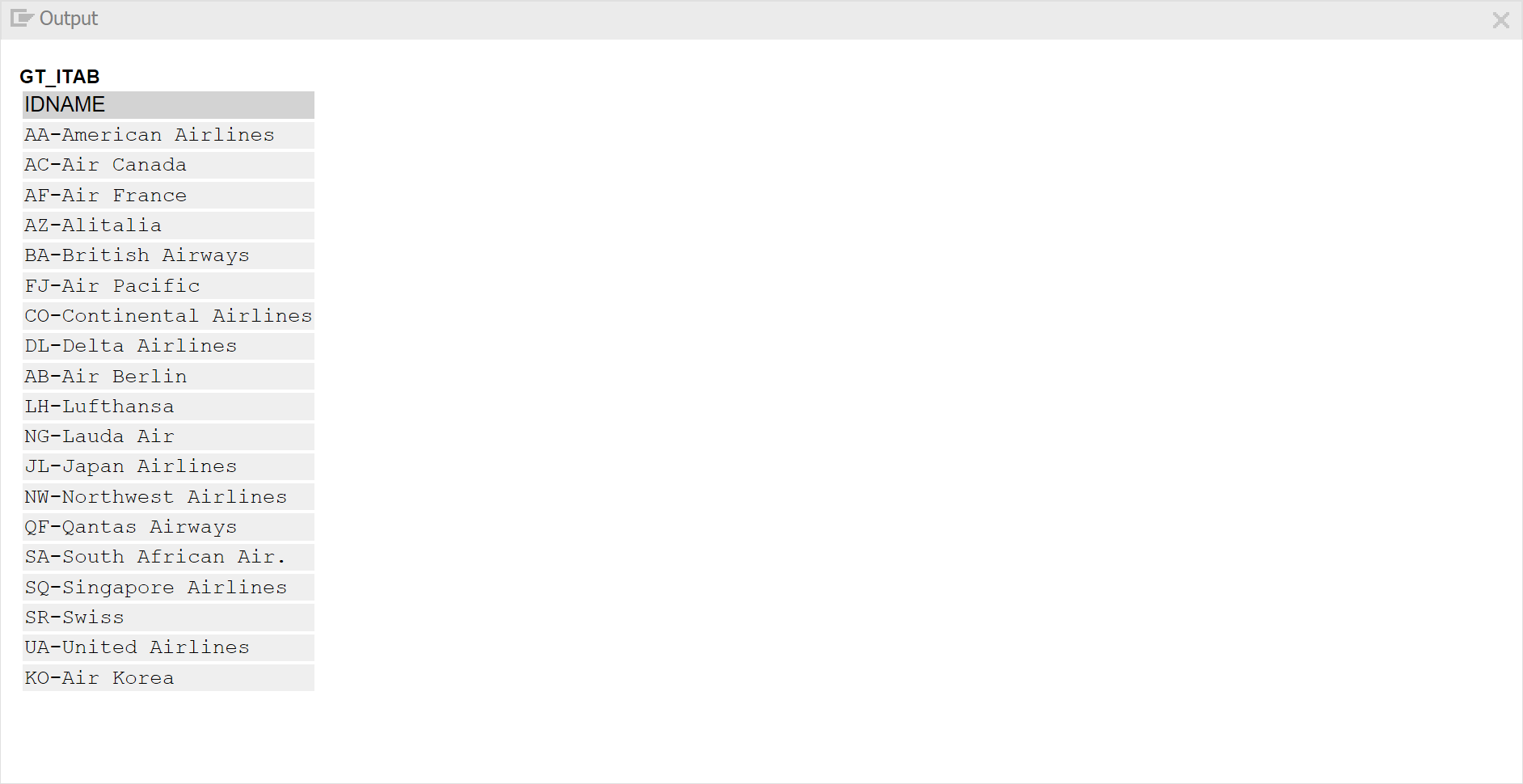
SE80 | 0504 | NEW SQL QUERY | CAST 옵션 사용

\*CAST 옵션을 사용하여 컬럼 타입을 변환할 수 있다.  
TYPES : BEGIN OF T\_STR,  
               CARRID TYPE SFLIGHT-CARRID,  
               CONNID TYPE SFLIGHT-CONNID,  
               OCCUP TYPE F,  
END OF T\_STR.  
  
DATA: GT\_ITAB TYPE TABLE OF T\_STR.  
  
SELECT CARRID, CONNID,  CAST( SEATSOCC AS FLTP )  / CAST( SEATSMAX AS FLTP ) AS OCCUP  
               INTO CORRESPONDING FIELDS OF TABLE @GT\_ITAB  
               FROM SFLIGHT  
               WHERE SEATSOCC > 0.  
  
CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ) .



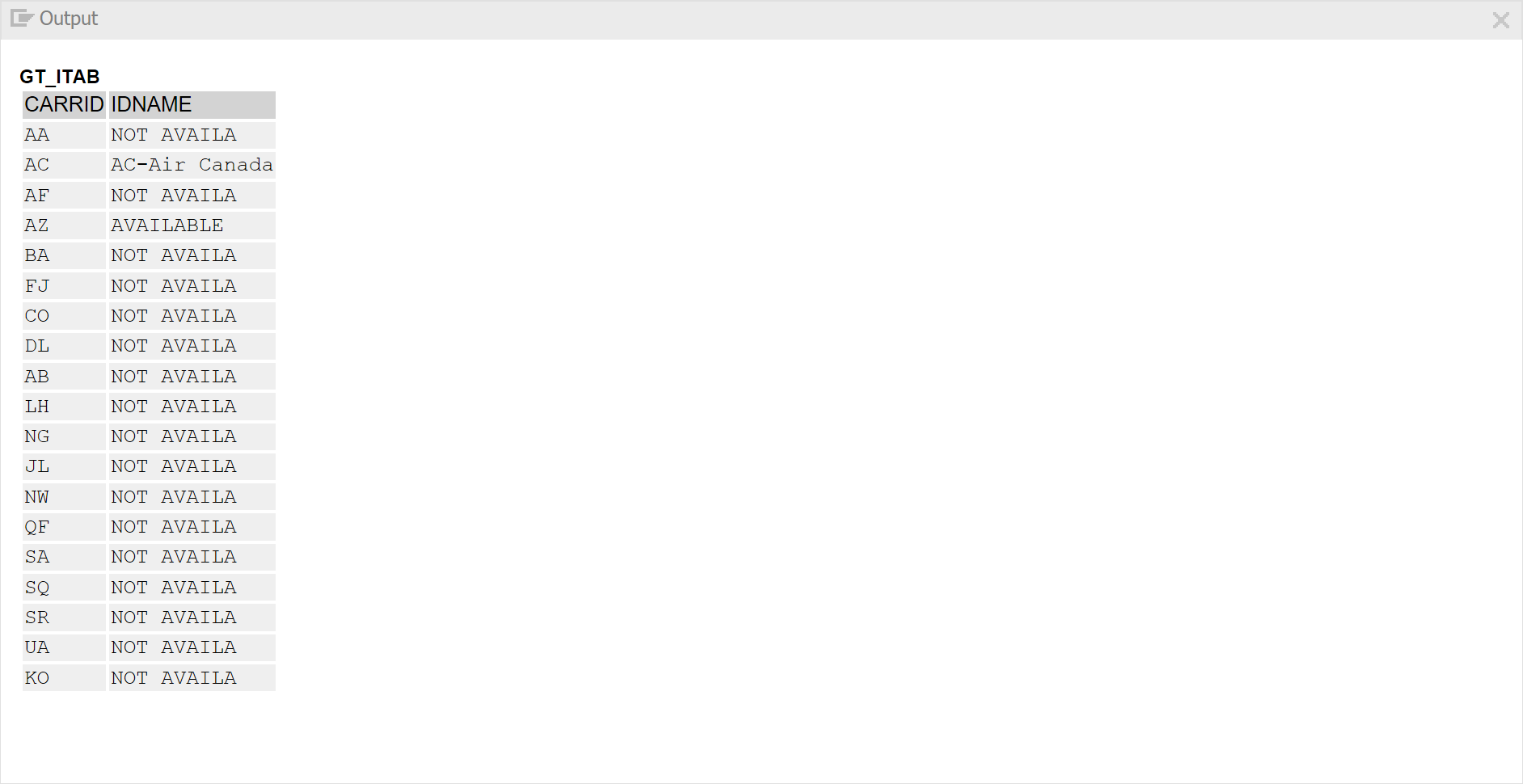
SE80 | 0505 | NEW SQL QUERY | && 사용하여 문자열 연결

\*&&을 사용하여 문자열을 연결할 수 있다.  
\*이때 연결된 문자열의 길이는 255자를 넘을 수 없다.  
  
TYPES: BEGIN OF T\_STR,  
             IDNAME TYPE C LENGTH 40,  
END OF T\_STR.  
  
DATA: GT\_ITAB TYPE TABLE OF T\_STR.  
  
SELECT CARRID && '-' && CARRNAME AS IDNAME  
              FROM SCARR  
              INTO CORRESPONDING FIELDS OF TABLE @GT\_ITAB.  
  
  CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ).



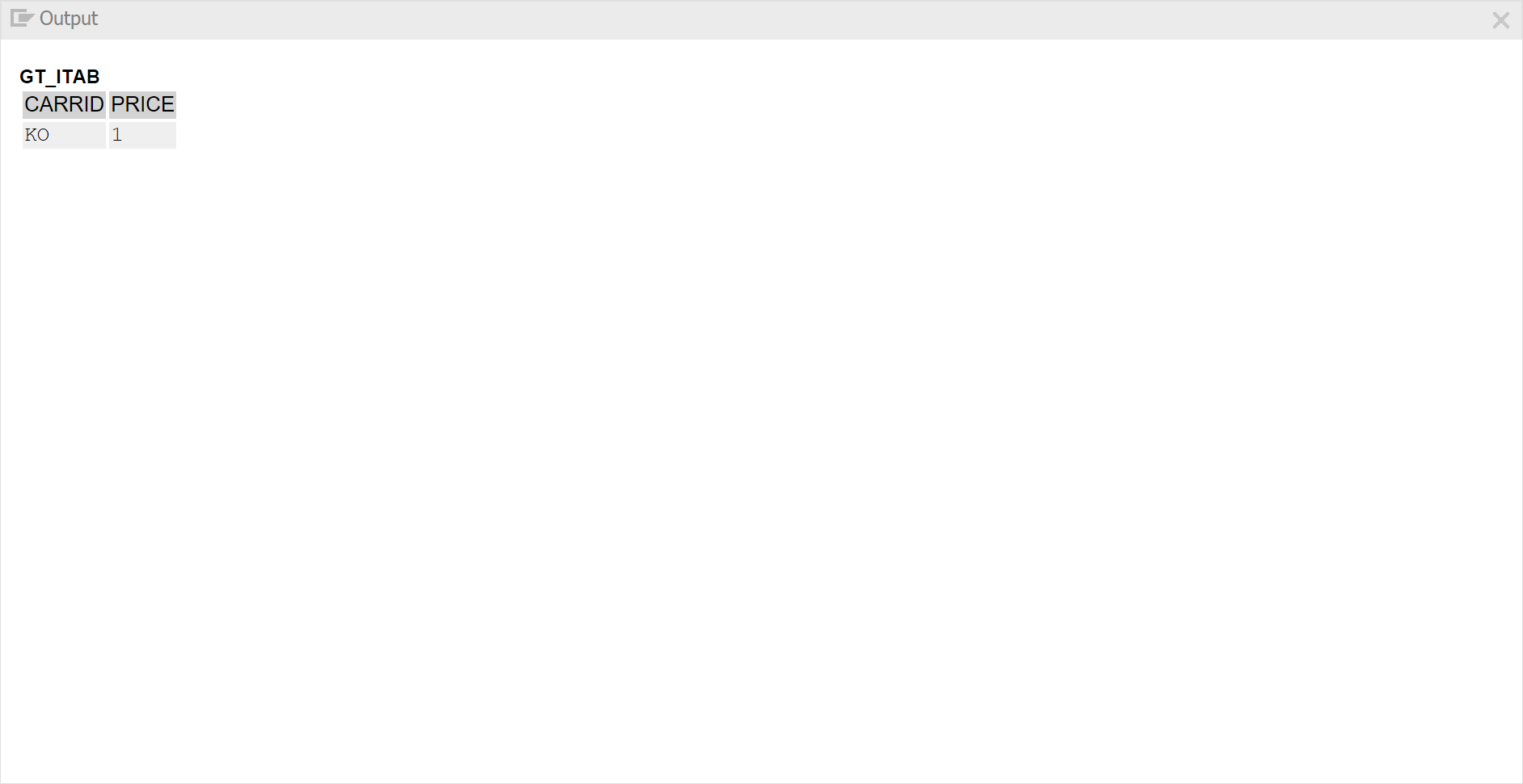
SE80 | 0506 | NEW SQL QUERY | CASE 구문 이용

\*CASE 구문을 이용하여 컬럼 값에 따라서 분기문으로 처리할 수 있다.  
  
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.  
  
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 TABLE @GT\_ITAB.  
  
  CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ).



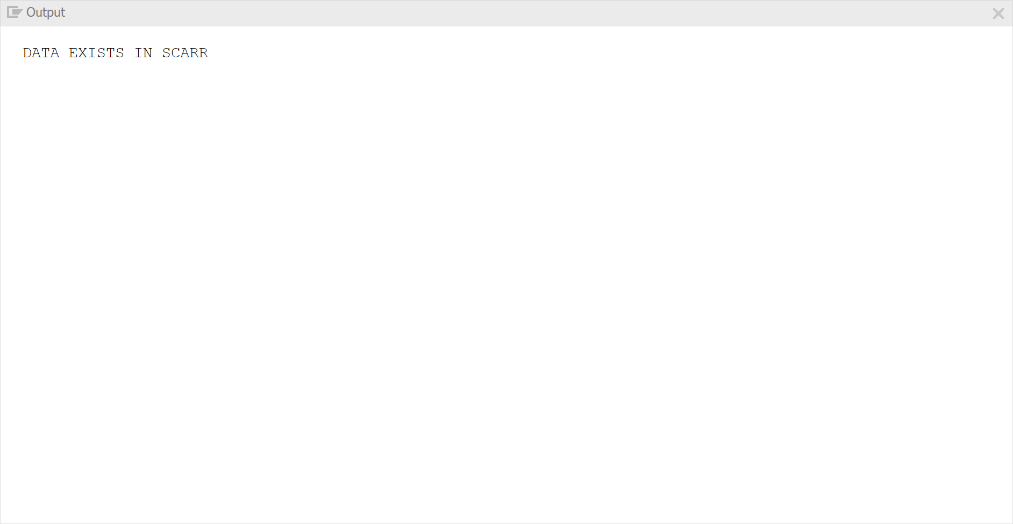
SE80 | 0507 | NEW SQL QUERY | COALESCE 사용하기

\*COALESCE 를 사용하여 NULL을 대체  
\*COALESCE ( ARG1, ARG2 )  
\*ABAP에서 DATA 선언 시 기본으로 INITAL VALUE로 지정된다.  
\*NULL은 사용되지 않는다고 간주해도 무방하다.  
\*COALESCE 구문은 OUTER JOIN을 할 경우 값이 없는 경우에 활용할 수 있다.  
TYPES: BEGIN OF T\_STR,  
             CARRID TYPE SCARR-CARRID,  
             PRICE TYPE I,  
END OF T\_STR.  
  
DATA: GT\_ITAB TYPE TABLE OF T\_STR.  
\*NULL 일 경우 1로 표시하자  
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'.  
CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ).



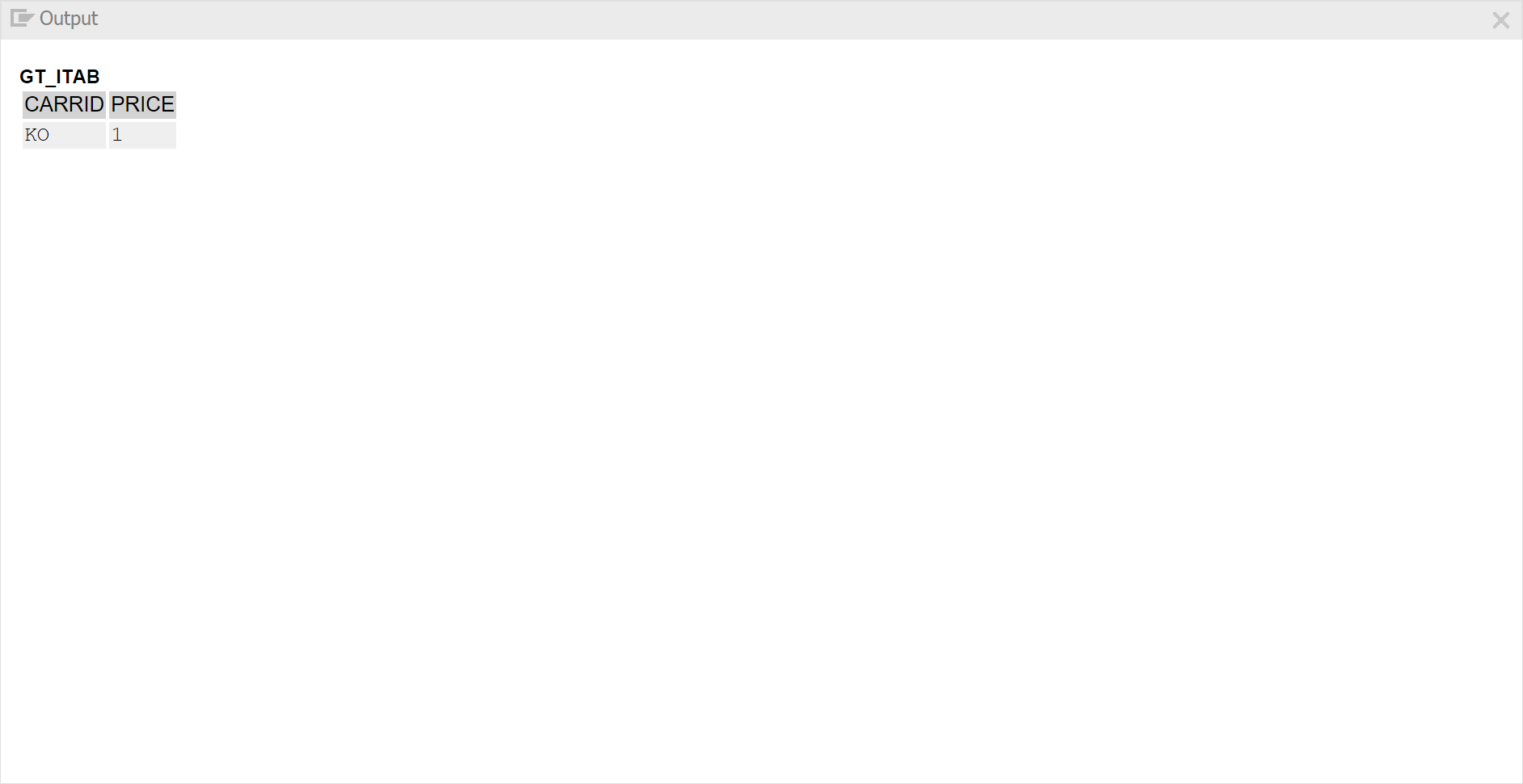
SE80 | 0508 | NEW SQL QUERY | CONSTANT VALUE

\*CONSTANT VALUE 사용  
\*조건에 맞는 데이터가 존재하는지 체크 할 경우 유용하게 사용할 수 있다.  
DATA: GV\_EXIST TYPE C.  
  
SELECT SINGLE @ABAP\_TRUE  
               FROM SCARR  
               WHERE CARRID = 'AC'  
               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.



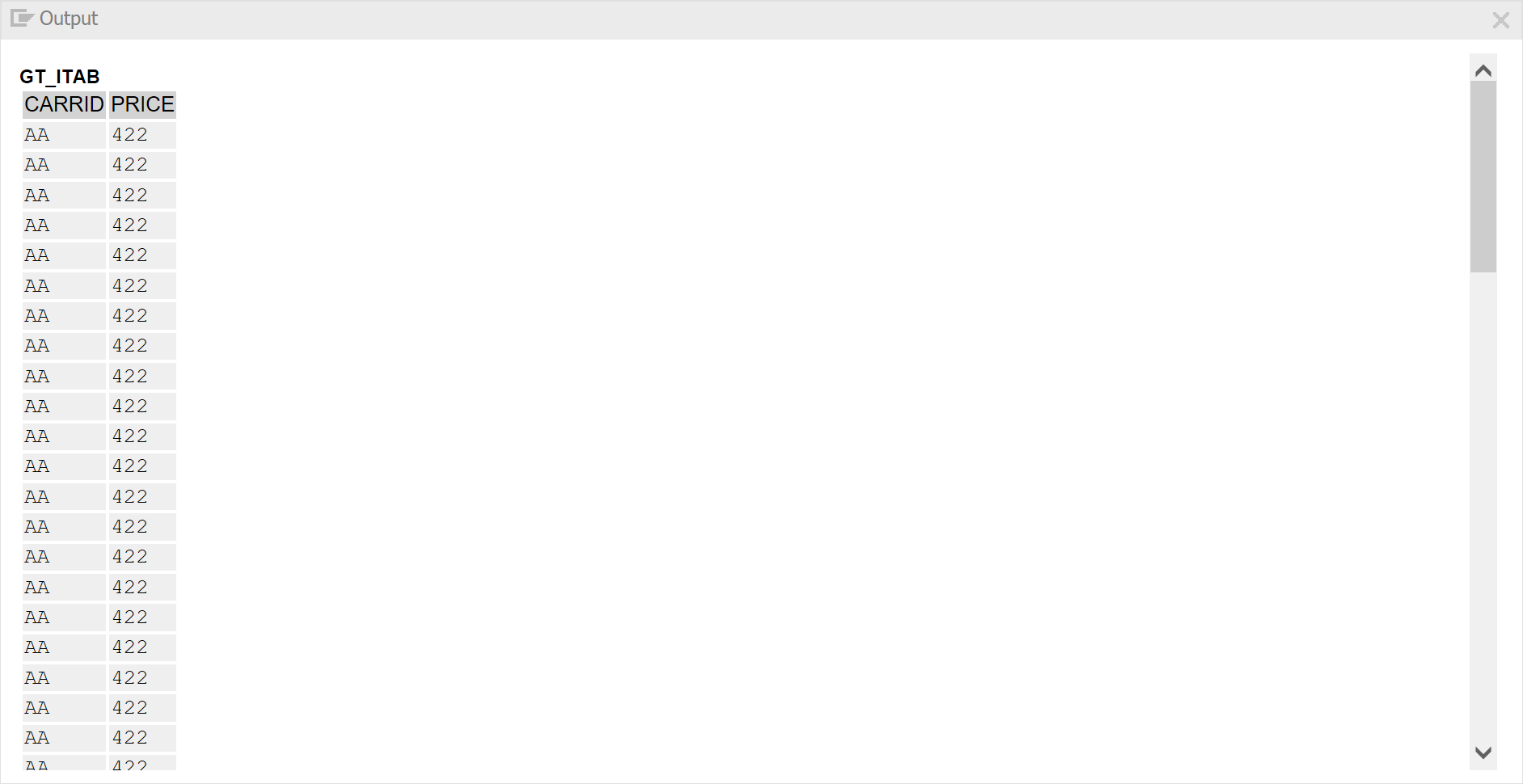
SE80 | 0509 | NEW SQL QUERY | RIGH OUTER JOIN

\*RIGHT OUTER JOIN 지원  
\*LEFT / RIGHT OUTER JOIN 이 모두 지원된다.  
\*WHERE 조건에  OUTER JOIN으로 사용된 칼럼이 조건 채 포함될 수 있다.  
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 SFLIGHT RIGHT OUTER JOIN SCARR  
              ON SFLIGHT~CARRID = SCARR~CARRID  
              INTO CORRESPONDING FIELDS OF TABLE @GT\_ITAB  
              WHERE SCARR~CARRID = 'KO'.  
  
CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ).



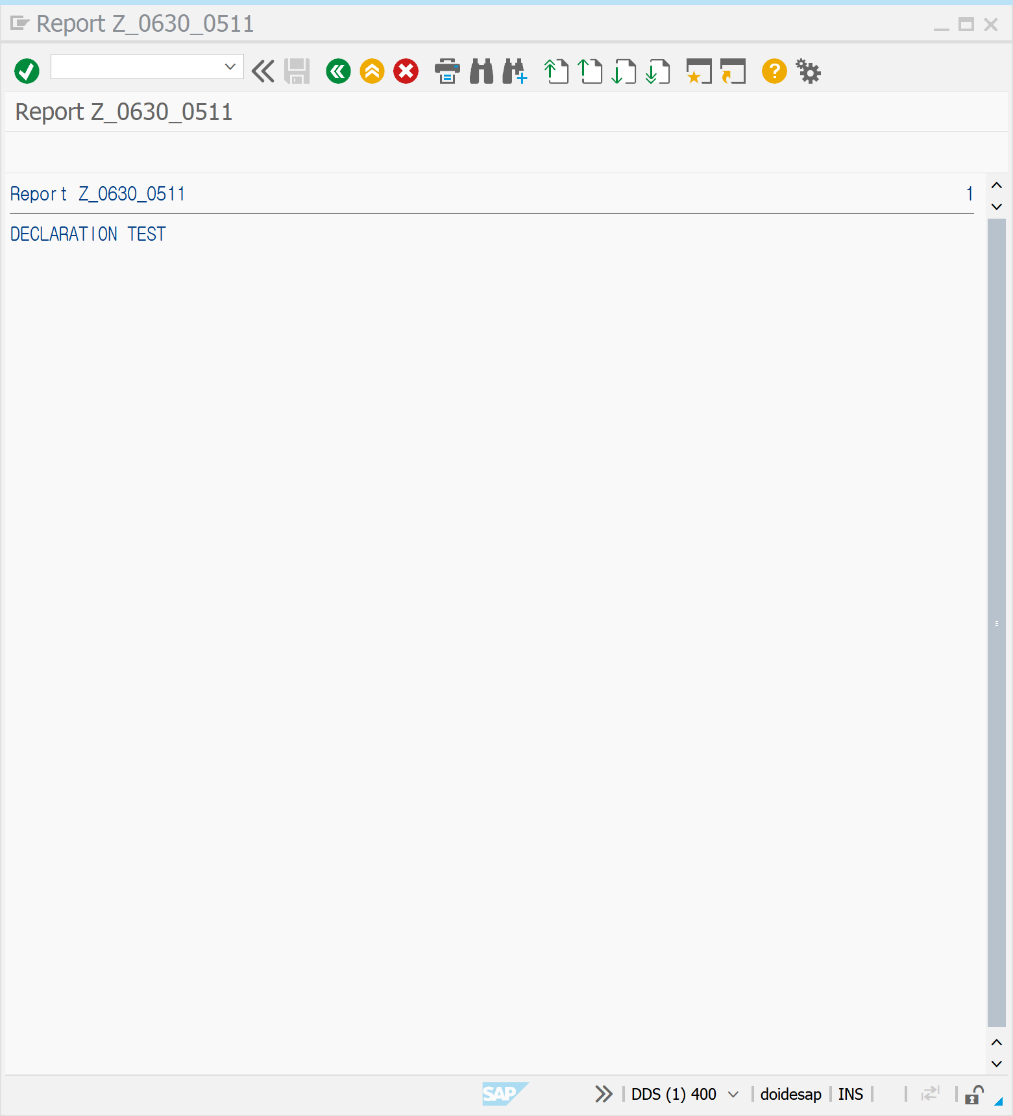
SE80 | 0510 | NEW SQL QUERY | JOIN에 사용하는 ON 조건 사용

\*JOIN에 사용되는 ON 조건 기능 확장  BETWEEN <> 사용사능  
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.  
  
CL\_DEMO\_OUTPUT=>DISPLAY( GT\_ITAB ).



SE80 | 0511 | NEW SQL QUERY | DECLARATION EXPRESSION

\*DECLARATION EXPRESSION은 변수를 선언하면서 동시에 값을 할당할 수 있다.  
\*DATA(VAR)  
  
DATA(GV\_VAR) = 'DECLARATION TEST'.  
WRITE: GV\_VAR.



SE80 | 0512 | 실습 예제를 확인해보자

\*284 144 2020.01.09  
TYPES: BEGIN OF GTY\_S\_BOOKING,  
             CARRID TYPE SBOOK-CARRID,  
             CONNID TYPE SBOOK-CONNID,  
             FLDATE TYPE SBOOK-FLDATE,  
             BOOKID TYPE SBOOK-BOOKID,  
             CUSTOMID TYPE SBOOK-CUSTOMID,  
             AGENCYNUM TYPE SBOOK-AGENCYNUM,  
            CARRNAME TYPE SCARR-CARRNAME,  
            AGENCYNAME TYPE STRAVELAG-NAME,  
            AGENCYCITY TYPE STRAVELAG-CITY,  
END OF GTY\_S\_BOOKING.  
  
TYPES: GTY\_T\_BOOKINGS TYPE STANDARD TABLE OF GTY\_S\_BOOKING  
                                              WITH NON-UNIQUE KEY  
                                              CARRID CONNID FLDATE BOOKID.  
  
DATA:  
      GT\_BOOKINGS TYPE GTY\_T\_BOOKINGS,  
      GS\_BOOKING TYPE GTY\_S\_BOOKING.  
  
DATA:  
      GV\_CUSTNAME TYPE SCUSTOM-NAME.  
\*      GV\_CARRNAME TYPE SCARR-CARRNAME,  
\*      GV\_AGENCYNAME TYPE STRAVELAG-NAME,  
\*      GV\_AGENCYCITY TYPE STRAVELAG-CITY.  
  
FIELD-SYMBOLS :  
               <FS\_BOOKING> LIKE LINE OF GT\_BOOKINGS.  
  
SELECT-OPTIONS :  
               SO\_AGY FOR GS\_BOOKING-AGENCYNUM DEFAULT '100',  
               SO\_CUS FOR GS\_BOOKING-CUSTOMID,  
               SO\_FLD FOR GS\_BOOKING-FLDATE.  
  
START-OF-SELECTION.  
              SELECT B~CARRID B~CONNID B~FLDATE B~BOOKID  
                            B~CUSTOMID B~AGENCYNUM  
                            C~CARRNAME A~NAME A~CITY  
                            FROM SBOOK AS B INNER JOIN SCARR AS C  
                            ON B~CARRID = C~CARRID  
                            LEFT OUTER JOIN STRAVELAG AS A  
                                      ON B~AGENCYNUM = A~AGENCYNUM  
                INTO TABLE GT\_BOOKINGS  
                WHERE B~AGENCYNUM IN SO\_AGY AND  
                              B~CUSTOMID IN SO\_CUS AND  
                              B~FLDATE IN SO\_FLD AND  
                              CANCELLED <> 'X'.  
LOOP AT GT\_BOOKINGS ASSIGNING <FS\_BOOKING>.  
  
\*SCUSTOM IS BUFFERED - NO NEDD FOR OPTIMIZATIONS  
  SELECT SINGLE NAME FROM SCUSTOM INTO GV\_CUSTNAME  
    WHERE ID = <FS\_BOOKING>-CUSTOMID.  
\* SELECT SINGLE CARRNAME FROM SCARR  
\* INTO GV\_CARRNAME  
\* WHERE CARRID = <fs.booking>-carrid.  
  
\*SELECT SINGLE name city FROM stravelag.  
\* INTO (gv\_agencyname, gv\_agencycity)  
\* WHERE agnecynum = <fs.booking>-agencynum.  
\*  
  
WRITE:/ <FS\_BOOKING>-CARRID,  
\*         gv\_carrname,  
        <FS\_BOOKING>-CARRNAME,  
        <FS\_BOOKING>-CONNID,  
        <FS\_BOOKING>-FLDATE,  
        <FS\_BOOKING>-BOOKID,  
        GV\_CUSTNAME,  
\*        gv\_agencyname,  
\*        gv\_agencycity,  
        <FS\_BOOKING>-AGENCYNAME,  
        <FS\_BOOKING>-AGENCYCITY.  
ENDLOOP.

