Index Notation for Multiple-Value Fields and Periodic Groups

This chapter describes how you can use the index notation (*n*:*n*) to specify how many values of a multiple-value field or how many occurrences of a periodic group are to be output.

The following topics are covered:

- Use of Index Notation

- Example of Index Notation in DISPLAY Statement

- Example of Index Notation in WRITE Statement

Use of Index Notation

With the index notation (*n*:*n*) you can specify how many values of a multiple-value field or how many occurrences of a periodic group are to be output.

For example, the field INCOME in the DDM EMPLOYEES is a periodic group which keeps a record of the annual incomes of an employee for each year he/she has been with the company.

These annual incomes are maintained in chronological order. The income of the most recent year is in occurrence 1.

If you wanted to have the annual incomes of an employee for the last three years displayed - that is, occurrences 1 to 3 - you would specify the notation (1:3) after the field name in a DISPLAY or WRITE statement (as shown in the following example program).

Example of Index Notation in DISPLAY Statement

```
** Example 'DISPLX07': DISPLAY (with index notation)
************************************************************************
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 BIRTH
  2 INCOME  (1:3)
    3 CURR-CODE
    3 SALARY
    3 BONUS (1:1)
END-DEFINE
*
READ (3) VIEWEMP BY BIRTH
  DISPLAY PERSONNEL-ID NAME INCOME (1:3)
  SKIP 1
END-READ
END
```

Output of Program DISPLX07:

Note that a DISPLAY statement outputs multiple values of a multiple-value field underneath one another:

```
Page      1                                          04-11-11   14:15:54

PERSONNEL           NAME                        INCOME
  ID
                                     CURRENCY   ANNUAL    BONUS
                                       CODE     SALARY
--------- -------------------- -------- ---------- ----------

30020013  GARRET                       UKL           4200          0
                                       UKL           4150          0
                                                        0          0

30016112  TAILOR                       UKL           7450          0
                                       UKL           7350          0
                                       UKL           6700          0

20017600  PIETSCH                      USD          22000          0
                                       USD          20200          0
                                       USD          18700          0
```

As a WRITE statement displays multiple values horizontally instead of vertically, this may cause a line overflow and a - possibly undesired - line advance.

If you use only a single field within a periodic group (for example, SALARY) instead of the entire periodic group, and if you also insert a slash (/) to cause a line advance (as shown in the following example between NAME and JOB-TITLE), the report format becomes manageable.

Example of Index Notation in WRITE Statement

```
** Example 'WRITEX03': WRITE (with index notation)
************************************************************************
DEFINE DATA LOCAL
1 VIEWEMP VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 BIRTH
  2 JOB-TITLE
  2 SALARY (1:3)
END-DEFINE
*
READ (3) VIEWEMP BY BIRTH
  WRITE PERSONNEL-ID NAME / JOB-TITLE SALARY (1:3)
  SKIP 1
END-READ
END
```

Output of Program WRITEX03:

```
Page      1                                                      04-11-11  14:15:55

30020013 GARRET
TYPIST                               4200        4150           0

30016112 TAILOR
WAREHOUSEMAN                         7450        7350        6700

20017600 PIETSCH
SECRETARY                           22000       20200       18700
```

```
Page      1                                                      04-11-11  14:15:55

30020013 GARRET
TYPIST                               4200        4150           0
```