

AT END OF DATA

Structured Mode Syntax

```
[AT]END [OF] DATA [(r)]
    statement ...
END-ENDDATA
```

Reporting Mode Syntax

```
[AT]END [OF] DATA [(r)]
{
    statement
    DO statement ... DOEND
}
```

This chapter covers the following topics:

- Function
- Restrictions
- Syntax Description
- Example

For an explanation of the symbols used in the syntax diagram, see *Syntax Symbols*.

Related Statements: ACCEPT/REJECT | AT BREAK | AT START OF DATA | BACKOUT TRANSACTION | BEFORE BREAK PROCESSING | DELETE | END TRANSACTION | FIND | GET | GET SAME | GET TRANSACTION DATA | HISTOGRAM | LIMIT | PASSW | PERFORM BREAK PROCESSING | READ | RETRY | STORE | UPDATE

Belongs to Function Group: *Database Access and Update*

Function

The AT END OF DATA statement is used to specify processing to be performed when all records selected for a database processing loop have been processed.

This section covers the following topics:

- Processing
- Values of Database Fields

- Positioning
- System Functions

See also *AT START/END OF DATA Statements* in the *Programming Guide*.

Processing

This statement is non-procedural, that is, its execution depends on an event, not on where in a program it is located.

Values of Database Fields

When the AT END OF DATA condition for the processing loop occurs, all database fields contain the data from the last record processed.

Positioning

This statement must be specified within the same program module which contains the loop creating statement.

System Functions

Natural system functions may be used in conjunction with an AT END OF DATA statement as described in *Using System Functions in Processing Loops* in the *System Functions* documentation.

Restrictions

- This statement can only be used in a processing loop that has been initiated with one of the following statements: FIND, READ, READ WORK FILE, HISTOGRAM or SORT.
- It may be used only once per processing loop.
- It is *not* evaluated if the processing loop referenced for END OF DATA processing is not entered.

Syntax Description

<i>(r)</i>	<p>Reference to a Specific Processing Loop: An AT END OF DATA statement may be related to a specific active processing loop by using the notation (<i>r</i>). If this notation is not used, the AT END OF DATA statement will be related to the outermost active database processing loop.</p>
END-ENDDATA	The Natural reserved word END-ENDDATA must be used to end the AT END OF DATA statement.

Example

```
** Example 'AEDEX1S': AT END OF DATA
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
```

```

2 PERSONNEL-ID
2 NAME
2 FIRST-NAME
2 SALARY (1)
2 CURR-CODE (1)
END-DEFINE
*
LIMIT 5
EMP. FIND EMPLOY-VIEW WITH CITY = 'STUTTGART'
IF NO RECORDS FOUND
  ENTER
END-NOREC
DISPLAY PERSONNEL-ID NAME FIRST-NAME
      SALARY (1) CURR-CODE (1)
/*
AT END OF DATA
IF *COUNTER (EMP.) = 0
  WRITE 'NO RECORDS FOUND'
  ESCAPE BOTTOM
END-IF
WRITE NOTITLE / 'SALARY STATISTICS:'
      / 7X 'MAXIMUM:' MAX(SALARY(1)) CURR-CODE (1)
      / 7X 'MINIMUM:' MIN(SALARY(1)) CURR-CODE (1)
      / 7X 'AVERAGE:' AVER(SALARY(1)) CURR-CODE (1)
END-ENDDATA
/*
END-FIND
*
END

```

See also *Natural System Functions for Use in Processing Loops*.

Output of Program AEDEX1S:

PERSONNEL ID	NAME	FIRST-NAME	ANNUAL SALARY	CURRENCY CODE
11100328	BERGHAUS	ROSE	70800	DM
11100329	BARTHEL	PETER	42000	DM
11300313	AECKERLE	SUSANNE	55200	DM
11300316	KANTE	GABRIELE	61200	DM
11500304	KLUGE	ELKE	49200	DM

SALARY STATISTICS:

MAXIMUM:	70800 DM
MINIMUM:	42000 DM
AVERAGE:	55680 DM

Equivalent reporting-mode example: AEDEX1R.