

AT START OF DATA

Structured Mode Syntax

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

Reporting Mode Syntax

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

This chapter covers the following topics:

- Function
- Syntax Description
- Example

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

Related Statements: ACCEPT/REJECT | AT BREAK | AT END 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 statement AT START OF DATA is used to perform processing immediately after the first of a set of records is read for a processing loop that has been initiated by one of the following statements: READ, FIND, HISTOGRAM, SORT or READ WORK FILE.

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

Processing

If the loop-initiating statement contains a WHERE clause, the at-start-of-data condition will be true when the first record is read which meets both the basic search and the WHERE criteria.

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

Value of Database Fields

All database fields contain the values of the record which caused the at-start-of-data condition to be true (that is, the first record of the set of records to be processed).

Positioning

This statement must be positioned *within* a processing loop, and it may be used only once per processing loop.

Syntax Description

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

Example

```

** Example 'ASDEX1S': AT START OF DATA (structured mode)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
  2 FIRST-NAME
  2 CITY
*
1 #CNTL (A1) INIT <' '>
1 #CITY (A20) INIT <' '>
END-DEFINE
*
REPEAT
  INPUT 'ENTER VALUE FOR CITY' #CITY
  IF #CITY = ' ' OR = 'END'
    STOP
  END-IF
  FIND EMPLOY-VIEW WITH CITY = #CITY
  IF NO RECORDS FOUND
    WRITE NOTITLE NOHDR 'NO RECORDS FOUND'
    ESCAPE BOTTOM
  END-NOREC
/*
  AT START OF DATA
    INPUT (AD=0) 'RECORDS FOUND' *NUMBER //
      'ENTER 'D'' TO DISPLAY RECORDS' #CNTL (AD=A)
    IF #CNTL NE 'D'
      ESCAPE BOTTOM
    END-IF
  END-START

```

```

/*
  DISPLAY NAME FIRST-NAME
END-FIND
END-REPEAT
END

```

Output of Program ASDEX1S:

ENTER VALUE FOR CITY **PARIS**

After entering and confirming name of city:

RECORDS FOUND 26

ENTER 'D' TO DISPLAY RECORDS **D**

Records displayed:

NAME	FIRST-NAME

MAIZIERE	ELISABETH
MARX	JEAN-MARIE
REIGNARD	JACQUELINE
RENAUD	MICHEL
REMOUE	GERMAINE
LAVENDA	SALOMON
BROUSSE	GUY
GIORDA	LOUIS
SIECA	FRANCOIS
CENSIER	BERNARD
DUC	JEAN-PAUL
CAHN	RAYMOND
MAZUY	ROBERT
FAURIE	HENRI
VALLY	ALAIN
BRETON	JEAN-MARIE
GIGLEUX	JACQUES
KORAB-BRZOZOWSKI	BOGDAN
XOLIN	CHRISTIAN
LEGRIS	ROGER
VVVV	

Equivalent reporting-mode example: ASDEX1R.