

POS - Field Identification Function

Format/length:

I4

The system function `POS(field-name)` contains the internal identification of the field whose name is specified with the system function.

`POS(field-name)` may be used to identify a specific field, regardless of its position in a map. This means that the sequence and number of fields in a map may be changed, but `POS(field-name)` will still uniquely identify the same field. With this, for example, you need only a single `REINPUT` statement to make the field to be `MARKed` dependent on the program logic.

Example:

```
DECIDE ON FIRST VALUE OF ...
  VALUE ...
    COMPUTE #FIELDX = POS(FIELD1)
  VALUE ...
    COMPUTE #FIELDX = POS(FIELD2)
  ...
END-DECIDE
...
REINPUT ... MARK #FIELDX
```

If the field specified with `POS` is an array, a specific occurrence must be specified; for example, `POS(FIELDX(5))`. `POS` cannot be applied to an array range.

POS and *CURS-FIELD

The system function `POS(field-name)` may be used in conjunction with the Natural system variable `*CURS-FIELD` to make the execution of certain functions dependent on which field the cursor is currently positioned in.

`*CURS-FIELD` contains the internal identification of the field in which the cursor is currently positioned; it cannot be used by itself, but only in conjunction with `POS(field-name)`. You may use them to check if the cursor is currently positioned in a specific field and have processing performed depending on that condition.

Example:

```
IF *CURS-FIELD = POS(FIELDX)
  MOVE *CURS-FIELD TO #FIELDY
END-IF
...
REINPUT ... MARK #FIELDY
```

Notes:

1. The values of `*CURS-FIELD` and `POS(field-name)` serve only as internal identifications of the fields and cannot be used for arithmetic operations.
2. The value returned by `POS(field-name)` for an occurrence of an X-array (an array for which at least one bound in at least one dimension is specified as expandable) may change after the number of occurrences for a dimension of the array has been changed using the `EXPAND`, `RESIZE` or `REDUCE` statements.

3. Natural RPC: If *CURS-FIELD and POS(*field-name*) refer to a context variable, the resulting information can only be used within the same conversation.
4. In Natural for Ajax applications, *CURS-FIELD identifies the operand that represents the value of the control that has the input focus. You may use *CURS-FIELD in conjunction with the POS function to check for the control that has the input focus and perform processing depending on that condition.

See also *Dialog Design, Field-Sensitive Processing and Simplifying Programming* in the *Programming Guide* in the *Programming Guide*.