

SAP ABAP 7.4 Syntax

Contents

SAP ABAP 7.4 Syntax	1
Inline Declarations.....	1
Table Expressions	3
Value Operator.....	5
For Operator.....	6
Switch Operator	6
Conditional Operator.....	7
Conversion Operator	7
Reduce Operator	7
New Operator.....	7
Strings	8
References	8

Inline Declarations

Variables Assignment

```
DATA(lv_num) = 1.
DATA(lv_num2) = lv_num.
```

Before 7.4

```
DATA: lv_num TYPE I,
      lv_num2 type I
lv_num = 1.
lv_num2 = lv_num.
```

Looping on Internal Tables

7.4 Syntax for loop into Work area:

```
LOOP AT itab INTO DATA(wa).
    "logic
ENDLOOP.
```

Example:

```
LOOP AT it_ekko INTO DATA(ls_ekko).
    "Logic
ENDLOOP.
```

Before 7.4 Syntax
DATA: ls_ekko TYPE ekko.
LOOP AT it_ekko INTO ls_ekko.
"Logic
ENDLOOP.

7.4 Syntax for loop at assigning:

```
LOOP AT itab ASSIGNING FIELD-SYMBOL(<fs_line>).  
  "logic  
ENDLOOP.
```

Example:

```
LOOP AT it_ekko ASSIGNING FIELD-SYMBOL(<fs_ekko>).  
  "Logic  
ENDLOOP.
```

Before 7.4 Syntax
FIELD-SYMBOLS <fs_ekko> type ekko.
LOOP AT it_ekko ASSIGNING <fs_ekko>.
"Logic
ENDLOOP.

Actual Parameters in Methods

After 7.4 Syntax
method1(
EXPORTING iv_input = lv_inp
IMPORTING iv_result = DATA(lv_result)).

Before 7.4 Syntax
DATA: lv_result TYPE char5.
method1(
EXPORTING iv_input = lv_inp
IMPORTING iv_result = lv_result).

Read Table Statements**7.4 Syntax for Read table into work area:**

```
READ TABLE it_ekko INTO DATA(ls_wa) WITH KEY col1 = value col2 = value.  
READ TABLE it_ekko INTO DATA(ls_wa) INDEX indx_val.
```

Example:

```
READ TABLE it_ekko INTO DATA(ls_ekko) WITH KEY ebeln = '5000000001' bsart =  
'NB'.  
READ TABLE it_ekko INTO DATA(ls_ekko) INDEX 1.
```

Before 7.4 Syntax
DATA: ls_ekko TYPE ekko.
READ TABLE it_ekko INTO ls_ekko WITH KEY ebeln = '5000000001' bsart = 'NB'.
READ TABLE it_ekko INTO ls_ekko INDEX 1.

7.4 Syntax for Read Assigning:

```
READ TABLE itab ASSIGNING FIELD-SYMBOL(<fs_line>).
```

Example:

```
READ TABLE it_ekko ASSIGNING FIELD-SYMBOL(<fs_ekko1>)
                        WITH KEY ebeln = '5000000001' bsart = 'NB'.
```

```
READ TABLE it_ekko ASSIGNING FIELD-SYMBOL(<fs_ekko>) INDEX 1.
```

Before 7.4 Syntax

```
FIELD-SYMBOLS <fs_ekko> TYPE ekko.
```

```
READ TABLE it_ekko ASSIGNING <fs_ekko> WITH KEY ebeln = '5000000001'
                        bsart = 'NB'.
```

```
READ TABLE it_ekko ASSIGNING <fs_ekko> INDEX 1.
```

Selection Statements**7.4 Syntax for Select Statement:**

```
SELECT * FROM db_tab_name INTO TABLE @DATA(itab) WHERE field = @lv_field.
```

```
SELECT SINGLE f1,f2 FROM db_tab_name INTO @DATA(ls_str) WHERE
field = @lv_field.
```

Example:

```
SELECT * FROM ekko INTO TABLE @DATA(it_ekko) WHERE ebeln = @lv_ebeln.
SELECT SINGLE ebeln, ebelp FROM ekpo into @DATA(ls_ekpo) WHERE ebeln =
@lv_ebeln.
```

Before 7.4 Syntax

```
DATA: it_ekko TYPE TABLE OF ekko,
      ls_ekko TYPE ekko.
```

```
SELECT * FROM ekko INTO TABLE it_ekko WHERE ebeln = lv_ebeln.
```

```
SELECT SINGLE ebeln, ebelp FROM ekpo into @ls_ekpo WHERE ebeln = @lv_ebeln.
```

Table Expressions**7.4 Syntax for Read Table with Index and Components**

```
DATA(wa) = itab[ index ].
```

```
DATA(wa) = itab[ value ].
```

```
DATA(wa) = itab[ col1 = value col2 = value].
```

Example:

```
DATA(ls_ekko1) = it_ekko[ '5000000001' ].
```

```
DATA(ls_ekko2) = it_ekko[ ebeln = '5000000001' ].
```

```
DATA(ls_ekko2) = it_ekko[ ebeln = '5000000001' bsart = 'NB' ].
```

```
DATA(ls_ekko3) = it_ekko[ 1 ].
```

Before 7.4 Syntax

```
READ TABLE it_ekko INTO ls_ekko INDEX 1.
```

```
READ TABLE it_ekko INTO ls_ekko WITH KEY ebeln = '5000000001'.
```

```
READ TABLE it_ekko INTO ls_ekko WITH KEY ebeln = '5000000001' bsart = 'NB'.
```

7.4 Syntax for Line Exists in Table

```
IF line_exists( itab[ coll = value ] ).  
  "logic  
ENDIF.
```

Example:

```
IF line_exists( it_ekko[ ebeln = '5000000001' bsart = 'NB' ] ).  
  "logic  
ENDIF.
```

Before 7.4 Syntax

```
READ TABLE it_ekko INTO ls_ekko WITH KEY ebeln = '5000000001' bsart = 'NB'.  
IF sy-subrc EQ 0.  
  "logic  
ENDIF.
```

7.4 Syntax for Index of the record

```
DATA(lv_index) = line_index( itab[ col = value ] ).
```

Example:

```
DATA(lv_index) = line_index( it_ekko[ ebeln = '5000000001' ] ).
```

7.4 Syntax for Corresponding

```
DATA(wa2) = CORRESPONDING tb_name( wa1 ).
```

Example:

```
DATA(ls_ekko2) = CORRESPONDING ekko( ls_ekko ).
```

Before 7.4 Syntax

```
MOVE-CORRESPONDING ls_ekko to ls_ekko1.
```

7.4 Syntax for table assigning to another table

```
DATA: itab1 TYPE TABLE OF ekko.  
DATA(itab2) = itab1.
```

Example:

```
DATA(lt_tmp_table1) = it_ekko.
```

Before 7.4 Syntax

```
DATA: it_tmp_tb TYPE TABLE OF ekko,  
      it_ekko TYPE TABLE OF ekko.  
lt_tmp_tb = it_ekko.
```

7.4 Syntax to get particular field value of a table record

```
DATA(lv_field_var) = itab[ field = value ]-required_field_name
```

Example

```
DATA(ls_doc_type) = it_ekko[ ebeln = '5000000001' ]-bsart.
```

7.4 Syntax to check with table field

```
IF itab[ field = value ]- required_field_name = value.
```

```
ENDIF.
```

Example

```
IF it_ekko[ ebeln = '5000000001' ]-bsart = 'NB'.
```

```
ENDIF.
```

Value Operator

Add records to internal table

Example 1

```
APPEND VALUE #( ebeln = ls_ekpo-ebeln ebelp = ls_ekpo-ebelp werks = ls_ekpo-
werks matnr = ls_ekpo-matnr ) TO it_ekpo.
```

Example 2

```
DATA(lt_rg) = VALUE rseloption( ( sign = 'I' option = 'EQ' low = '1' ) ).
```

Example 3

```
APPEND VALUE #( ebeln = ls_ekko-ebeln ) TO it_ekpo.
```

Add multiple records to internal table

Example 1

```
TYPES ty_ekko TYPE TABLE OF ekko WITH DEFAULT KEY.
TYPES ty_ekpo TYPE TABLE OF ekpo WITH DEFAULT KEY.
DATA(it_ekko1) = VALUE ty_ekko( ( ebeln = '5000000001' ) ( ebeln = '500000000
2' ) ).
```

Example 2

```
DATA: it_ekpo TYPE TABLE OF ekpo,
      ls_ekpo TYPE ekpo.
it_ekpo = VALUE #( ( ebeln = '5000000001' ) ( ebeln = '5000000002' ) ( ebeln
= '5000000003' ) ).
```

Example 3

```
DATA lt_range TYPE TABLE OF rsdsselect.
lt_range = VALUE #( ( sign = 'I' option = 'EQ' low = '1' ) ( sign = 'I' opti
on = 'EQ' low = '2' ) ).
```

Add values to work area

```
DATA(ls_ekko1) = VALUE ekko( ebeln = '5000000001' ).
DATA(ls_ekko1) = VALUE #( it_ekko[ ebeln = '5000000001' ] OPTIONAL ).
```

Filter the internal table and pass records to another internal table

Example

```
it_ekpo1 = VALUE #( BASE it_ekpo FOR ls_ekko IN it_ekko ( ebeln = ls_ekko-
ebeln ) ).
```

7.4 Syntax to get total lines of internal table

```
DATA(lv_lines) = LINES( itab ).
```

Example

```
DATA(lv_lines) = LINES( it_ekko ).
```

7.4 Syntax to check record exist in table

```

SELECT SINGLE @abap_true FROM ekko INTO @DATA(lv_flag) WHERE ebeln = value.
IF lv_flag eq abap_true.
  "logic
ELSE.
  "logic
ENDIF.

```

For Operator**7.4 Syntax**

```
FOR wa|<fs> IN itab [INDEX INTO idx] [cond]
```

Example 1

```
TYPES ty_ekpo TYPE TABLE OF ekpo WITH DEFAULT KEY.
```

```
DATA(it_ekpo1) = VALUE ty_ekpo( FOR ls_ekpo IN it_ekpo WHERE ( ebeln = '5000
000001' ) ( ebeln = ls_ekpo-ebeln ) ).
```

Example 2

```
TYPES lty TYPE TABLE OF rsdsselopt WITH DEFAULT KEY.
```

```
DATA(lt_rg) = VALUE rselooption( FOR <fs> IN it_ekpo ( sign = 'I' option = 'EQ
' low = <fs>-ebeln ) ).
```

Example 3

```
DATA(lt_rg) = VALUE rselooption( FOR <fs> IN it_ekpo WHERE ( ebeln IS NOT INIT
IAL ) ( sign = 'I' option = 'EQ' low = <fs>-ebeln ) ).
```

Switch Operator**7.4 Syntax**

```

DATA(lv_var) = SWITCH dtype|#( operand
                                WHEN const1 THEN result1
                                WHEN const2 THEN result2
                                ELSE resultn ).

```

Example

```

DATA(lv_num2) = SWITCH #( p_value WHEN 'A' THEN 1
                             WHEN 'B' THEN 2
                             WHEN 'A' THEN 3 ).

```

Before 7.4 Syntax

```

DATA: p_value TYPE c VALUE 'A',
      lv_num TYPE i.

```

```

CASE p_value.
  WHEN 'A'.
    lv_num = 1.
  WHEN 'B'.
    lv_num = 2.
  WHEN 'C'.
    lv_num = 3.
ENDCASE.

```

Conditional Operator

7.4 Syntax

```
DATA(lv_val) = COND dtype( WHEN log_exp1 THEN result1
                           WHEN log_exp2 THEN result2
                           ELSE resultn ).
```

Example

```
DATA(time) =
  COND string(
    WHEN sy-timlo < '120000' THEN |{ sy-timlo TIME = ISO } AM|
    WHEN sy-timlo > '120000' THEN |{ CONV t( sy-timlo -
      12 * 3600 ) TIME = ISO } PM|
    WHEN sy-timlo = '120000' THEN |High Noon| ).
```

Conversion Operator

Example

```
DATA txt TYPE abap_bool.
DATA str TYPE string.
```

```
txt = ' '.
str = ` `.
```

```
IF txt = str.
  " This block is executed.
ENDIF.
```

```
IF txt = CONV abap_bool( str ).
  " This block is executed.
ENDIF.
```

Reduce Operator

7.4 Syntax

```
DATA(lv_result) = REDUCE dt_type( INIT result = start_value FOR exp1
                                  NEXT result = iterated_value )
```

Example

```
DATA(lines) = REDUCE i( INIT x = 0 FOR ls_ekko1 IN it_ekko
                       WHERE ( ebeln = '5000000000' )
                       NEXT x = x + 1 ).
```

New Operator

7.4 Syntax

```
DATA(lo_ref) = NEW class_name.
```

Example

```
DATA(lo_ref) = NEW zcl_class.
```

Before 7.4 Syntax

```
DATA lo_ref TYPE REF TO zcl_class.
CREATE OBJECT lo_ref.
```

Conversions

7.4 Syntax

```
DATA(lv_var) = |{ lv_var_field ALPHA = IN | OUT } |.
```

Example

```
DATA(lv_matnr_in) = |{ lv_matnr ALPHA = IN } |.  
DATA(lv_matnr1) = |{ lv_matnr_IN ALPHA = OUT } |.
```

Before 7.4 Syntax

```
DATA lv_matnr TYPE matnr.  
lv_matnr = '1001'.  
  
call FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'  
  EXPORTING  
    input = lv_matnr  
  IMPORTING  
    output = lv_matnr.
```

Text Translations

7.4 Syntax

```
DATA(lv_var) = | { lv_var_text CASE = LOWER | UPPER } |.
```

Example

```
DATA(lv_text_l) = | { lv_text CASE = LOWER } |.  
DATA(lv_text_u) = | { lv_text_l CASE = UPPER } |.
```

Before 7.4 Syntax

```
DATA lv_text(10) TYPE c VALUE 'SAP ABAP'.  
TRANSLATE lv_text TO LOWER CASE.  
TRANSLATE lv_text TO UPPER CASE.
```

Strings

Example

```
DATA(lv_string) = |HELLO| & |WORLD|.   
DATA(lv_string1) = |HELLO| & | | & |WORLD|.
```

Before 7.4 Syntax

```
DATA lv_output TYPE STRING.  
CONCATENATE 'HELLO' 'WORLD' INTO lv_output SEPARATED BY space.
```

References

https://help.sap.com/doc/abapdocu_latest_index_htm/latest/en-US/index.htm?file=abennews-740.htm