INTRODUCTION

SAP Script is the SAP System's own text-processing system. Also called as Word processing tool of SAP.

SAP Script is tightly integrated into the SAP System

Note:
Entering and formatting text in SAP script is a little bit different than traditional text-processing programs on a PC. The main difference: the text that you enter is not in its final format. SAP script applies your formatting to text only when a document is printed, print-previewed, or displayed on-line in the SAP System.
Sap Script

How does a SAP Script work?

Every Script has a driver / print program attached or calling it.

In case you are creating an entirely new script, you also need to create the driver program.

This program can be executed directly or attached to certain events viz. Creation of Purchase Order, Sales Order, Hiring an Employee.

Standard SAP scripts can be modified by copying them in the customer name space and making the required changes.
Components of a SAP Scripts

1. Standard Text
   This is like a normal document. You can create letters, articles basically any static text.
   This can be created from transaction SO10.

2. Layout Set
   This also called as Form created from Transaction SE71.

3. Print Program
   This program retrieves desired data from the database, calls the Form, supplies data to the Form and finally closes the Layout Set (Form).
Sap Script

Driver Program → Supplies data → Layout set

Sap Script → Defines

Gets printed → Forms
Sap Script

Name
House name
Postal Code
City

WINDOW

Page

Page window

Particular area in the page where the window is printed out
### Standard Text: Request

<table>
<thead>
<tr>
<th>Standard text</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Text name</td>
<td>Z_NDC_TRAINING</td>
<td></td>
</tr>
<tr>
<td>Text ID</td>
<td>ST</td>
<td>Standard text</td>
</tr>
<tr>
<td>Language</td>
<td>EN</td>
<td></td>
</tr>
</tbody>
</table>

**Transaction SO10**
After creating a Standard Text assign it a Style.
If you want text in your standard text to appear as per attributes of character string, you need to enclose the text `<character string>TEXT</`.

...+...1...+...2...+...3...+...4...+...5...+...6...+...7...

' <U>H: this is the first script for NDC</>
OBJECTS OF LAYOUT SET

• Header
• Paragraph and Character format
• Windows and Text Elements
• Pages
• Page Windows
Header data is found both in style and layout set maintenance. In style maintenance, it is used primarily to present information designed to make it easier for the end user to select a style.

The header data in layout set maintenance, on the other hand, is used for information and control purposes.
In SAP Script, paragraphs are formatted using formats and their corresponding attributes. Word processing is simplified by the use of different paragraph attribute groups:

- Standard
- Font
- Tabs
- Outline
Character Format

Character formats
• Allow you to format entire blocks of text within a paragraph
• Which will override the paragraph format
• Character formats are defined in character strings

Eg: <B1> &VBDKA-TDNAME& ,, &VBDKA-TFGEU&</>
Windows and Text Elements

WINDOWS

We can combine information contained in a document into certain groups and make each group appear on the printed page in an individual area. SAP Scripts calls such a group a window.

- To refer a window via program each window must have a unique name
- We can assign text to each window via text elements

TEXT ELEMENTS

Text elements can be defined in layout set maintenance for each window. The text elements are accessed by name by a print program, formatted, and output in the respective window. The /E in the tag column is used to identify the text as a text element.
Windows are defined in layout set maintenance. They represent areas which are positioned on pages -- as page windows -- and in which text is later output. At least one window must be defined for each layout set. If not, a text cannot be formatted by SAP Script.

The following window types can be used:
- main
- variable
- constant
Windows

MAIN
Main window in which continuous text is output. The text in the main window can extend over several pages. If the text fills one page, output continues in the window of the next and subsequent pages, as long as MAIN has been defined for these pages.

VAR
Window with variable contents. The text can vary on each page in which the window is positioned. Variable windows are formatted for each page.

CONST
Window with constant contents which is only formatted once.
At least one page must be defined for every layout set. You must also designate a "first" page in the layout set header. Otherwise text formatting is not possible.
### Form: Display Pages: ZTESTKRISH

<table>
<thead>
<tr>
<th>Page</th>
<th>Meaning</th>
<th>NextPage</th>
<th>Mode</th>
<th>No. type</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRO</td>
<td>first page</td>
<td>NEXT</td>
<td>INC</td>
<td>ARABIC</td>
</tr>
<tr>
<td>NEXT</td>
<td>next page</td>
<td>NEXT</td>
<td>INC</td>
<td>ARABIC</td>
</tr>
</tbody>
</table>

Page 1 of 2
Page Windows

When you define page windows, the window names are linked to page names. To do this, you need to specify the position and size of the window on the assigned page.

Width of the main window must be the same on all pages.
COMPONENTS OF SCRIPT

• Text Elements
• Style
• Layout set
The text elements are related to a window. The print program calls the respective text elements of the window.

**TYPES OF TEXT ELEMENTS**

- TEXT ELEMENTS WITH OUT NAMES
- TEXT ELEMENTS WITH NAME
Text Elements

NAMED TEXT ELEMENTS

• We recognize named text elements by paragraph format /E

• The name can be up to 40 characters

• The different windows can have same named text elements

• To output named text elements we have to call the function module WRITE_FORM
A style defines the set of paragraph and character formats that are available for formatting a document.

If a style is selected, then it provides the character and paragraph formats for your document. Any formats defined in the layout set selected for a document are then ignored.
• Layout sets are used to control the page layout and also the text formatting in your documents.

• If a document is formatted for output to the screen or to a printer, then a layout set must be specified for it.

• If no other layout set is specified for a document, then the SYSTEM layout set is assigned to the document by default.
LAYOUT SET CONTROL

OPEN_FORM  - Initializes the SAP script composer and opens the specified layout set
CLOSE_FORM - Closes the layout set (if close_form is missing in the program, nothing will be printed)
WRITE_FORM, WRITE_FORM_LINES, CONTROL_FORM - To output data (this can be used as many times between OPEN_FORM and CLOSE_FORM)

PAGE CONTROL IN LAYOUT SETS

• STATIC DEFINITION - Specify the starting page in header and for each page specify the subsequent page
• DYNAMIC DEFINITION - Specify START PAGE when you call START_FORM
Symbols

SYMBOLS

Symbols are constants that can be inserted in a document. They save the user unnecessary work when replacing sections of text, phrases, etc. Possible symbol types are:

• System symbols

• Standard symbols

• Program symbols

• Text symbols.
Symbols

THE SYNTAX OF SYMBOLS

• The delimiter & must be used both immediately before and after the symbol.

• The name of a symbol may not contain spaces.

• A SAP Script editor line break should not occur between the symbol delimiters.

• If additional formatting options are required, these should be enclosed in round brackets and inserted immediately after the symbol name.

• The code letters identifying these options must be given in capitals.
SYSTEM SYMBOLS

SAP script provides values for the system symbols
Example &date& &time& &month& &year& &uline& &vline& &page&

STANDARD SYMBOLS

• Standard symbols are defined in the TTDTG table.
• This table contains both the name of each symbol and its value.
• The value, which is language-dependent, can contain up to 60 chars.
• This table is delivered complete with standard entries supplied by SAP.
• It may be extended with customer-specific symbols
SAP Script cannot read this data out of the SAP database itself, but has to call on another program to do this. The data are then copied into work areas declared using TABLES. If SAP Script is now called from this program in order to format a text, it can copy the data out of these work areas.

Symbols which obtain their values from this kind of data area are called program symbols.

Note
• The value of a program symbol is limited up to a maximum of 255 chars.

• The name of a program symbol consists of the table name and the field name separated by a hyphen.
Symbols

TEXT SYMBOLS

All symbols which do not correspond to one of the three types of symbol described above are text symbols. You define the value of a text symbol yourself in the text module.

There are two ways of doing this:
- Use the menu entries in the text editor
- Use the control command DEFINE

for e.g

/: DEFINE &mysymbol& = ‘XXXX’
/ &mysymbol&
/: DEFINE &mysymbol& = ‘YYYY’
/ &mysymbol&

would print

XXXX
YYYY
Control Command is a type of SAP script command which are used to control output formatting.

The Syntax of Control Commands
• The paragraph format /: must be entered in the format column to identify a control.

• You enter the command itself in the text line.

• A control command, together with any parameters it requires, may not occupy more than a single line

• A maximum of one control command may appear in each line
Control Commands

SYNTAX OF CONTROL COMMANDS

• In tag column, indicate control commands with paragraph format /:

• Enter complete control commands and necessary parameters on one line

• Enter only one control command per line

• Editor formatting does not affect lines with control commands
SOME IMPORTANT CONTROL COMMANDS:

/: NEW-PAGE

/: PROTECT & /: ENDPROTECT

/: SET DATE MASK = 'date_mask'

/: SET TIME MASK = 'time_mask'

/: RESET paragraph_format

/: INCLUDE name [OBJECT o] [ID i] [LANGUAGE l]

[PARAGRAPH p] [NEW-PARAGRAPH np]

(Cont….. Next page)
Control Commands

/: TOP

/: ENDTOP

/: BOTTOM

/: ENDBOTTOM

/: IF condition

/: ENDF
SET DATE MASK command

The formatting for date fields can be defined with the SAPscript SET DATE MASK command. Executing this command causes all subsequent date fields to be output using the specified formatting.

Syntax
/: SET DATE MASK = 'date_mask'

The following templates may be used in the date mask:
For day : DD / DDD / DDDD
For month : MM / MMM / MMMM
For year : YY / YYYY
You can use the SAPscript SET TIME MASK command to format time fields in a way that differs from the standard setting. Executing this command causes all subsequent time fields to be output using the specified formatting.

Syntax:
/: SET TIME MASK = 'time_mask'

The following templates may be used in the time mask:
HH hours (two digits)
MM minutes (two digits)
SS seconds (two digits)
Control Commands

BOXES, LINES, SHADING: BOX, POSITION, SIZE

The BOX, POSITION and SIZE commands for drawing boxes, lines and shadowing can be used for specifying that within a layout set particular windows or passages of text within a window are to be output in a frame or with shadowing.

Syntax
1. /: BOX [XPOS] [YPOS] [WIDTH] [HEIGHT] [FRAME] [INTENSITY]
2. /: POSITION [XORIGIN] [YORIGIN] [WINDOW] [PAGE]
3. /: SIZE [WIDTH] [HEIGHT] [WINDOW] [PAGE]
ADDRESS

The command formats an address according to the postal standards of the destination country defined in the parameter COUNTRY. SAP scripts call function module ADDRESS_INTO_PRINTFORM to format the address.

/: ADDRESS[DELIVERY] [PARAGRAPH a] [PRIORITY p] [LINES l] /
/: TITLE title
/: NAME name1[,name 2[,name 3[,name 4]]]
/: STREET street
/: POBOX po box[CODE zip code]
/: CITY town1[,town 2]
/: REGION region
/: COUNTRY country
/: FROMCOUNTRY from country
/: ENDADDRESS
The text lines between the two commands are output at the bottom of MAIN window.

Text symbol receive their value through an explicit assignment. This assignment can be done interactively in the editor via include-symbols-text. The assigned value may have a maximum of 60 characters.
Incorporating Graphics

For e.g Company Logo

1. .bmp / .jpg file has to be converted to .tiff file

2. Upload the file as text module using ABAP/4 report “RSTXLDNC” specify BMON or BCOL

3. System generates a suggested name for the text e.g ZHEX-MACRO-LOGO Use ID “ST” and logon language.

4. Include ZHEX-MACRO-LOGO object Text ID ST in the Layout.
Example of Print Program

Calling an Script from a program created in SE38

REPORT ZSCRIPT.
TABLES: SPFLI.
DATA: I_SPFLI LIKE SPFLI OCCURS 0 WITH HEADER LINE.
SELECT * FROM SPFLI INTO CORRESPONDING FIELDS OF TABLE I_SPFLI.

CALL FUNCTION 'OPEN_FORM'
EXPORTING
FORM = 'ZTEST11'.

CALL FUNCTION 'WRITE_FORM'
EXPORTING
ELEMENT = 'HEADER_ITEM'
* FUNCTION = 'SET'
TYPE = 'BODY'
WINDOW = 'MAIN'.

(Contd..... Next Page)
CALL FUNCTION 'WRITE_FORM'  
EXPORTING  
 ELEMENT                        =  'ADDRESS_LIST'  
* FUNCTION                       = 'SET'  
 TYPE                           = 'BODY'  
 WINDOW                         = 'WINDOW1'.

LOOP AT I_SPFLI.  
  CALL FUNCTION 'WRITE_FORM'  
  EXPORTING  
 ELEMENT                        =  'LINE_ITEM'  
* FUNCTION                       = 'SET'  
 TYPE                           = 'BODY'  
 WINDOW                         = 'MAIN'.
  ENDLOOP.
CALL FUNCTION 'CLOSE_FORM'.

Example of Print Program
### Example of Print Program

**Output of the program**

**Satyam computers**
no 91 G.N Chetty
road
Madras

<table>
<thead>
<tr>
<th>CARRID</th>
<th>CONNID</th>
<th>TIME</th>
<th>CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA</td>
<td>0102</td>
<td>0:00</td>
<td>FRANKFURT</td>
</tr>
<tr>
<td>AA</td>
<td>0103</td>
<td>0:00</td>
<td>WELLINGTON</td>
</tr>
<tr>
<td>LH</td>
<td>0103</td>
<td>0:00</td>
<td>SYDNEY</td>
</tr>
<tr>
<td>LH</td>
<td>0104</td>
<td>0:00</td>
<td>FRANKFURT</td>
</tr>
<tr>
<td>AI</td>
<td>0106</td>
<td>0:00</td>
<td>AMSTERDUM</td>
</tr>
<tr>
<td>AI</td>
<td>0107</td>
<td>0:00</td>
<td>WELLINGTON</td>
</tr>
</tbody>
</table>
**Details of Layout Set Header Data**

*Form: Display Header: ZTEST11*

<table>
<thead>
<tr>
<th>Administrative data</th>
<th>Basic settings</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Form</strong></td>
<td>ZTEST11</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>Test program</td>
</tr>
<tr>
<td><strong>Status</strong></td>
<td>Active  Saved</td>
</tr>
<tr>
<td><strong>Classification</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Development class</strong></td>
<td>$TMP</td>
</tr>
<tr>
<td><strong>Client number</strong></td>
<td>555</td>
</tr>
<tr>
<td><strong>Created on</strong></td>
<td>09.01.2002 14:34:41 by DEVELOP</td>
</tr>
<tr>
<td><strong>Changed on</strong></td>
<td>10.01.2002 15:22:03 by DEVELOP</td>
</tr>
</tbody>
</table>

**Language attributes**

- **Language key**: EN
- **Original language**: EN

Translate
- to all languages
- to individual languages
- do not translate

..our people make the difference..
### Pages

<table>
<thead>
<tr>
<th>Page</th>
<th>Meaning</th>
<th>NextPage</th>
<th>Mode</th>
<th>No. type</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE1</td>
<td>New page</td>
<td>PAGE2</td>
<td>INC</td>
<td>ARABIC</td>
</tr>
<tr>
<td>PAGE2</td>
<td>New page</td>
<td>PAGE2</td>
<td>INC</td>
<td>ARABIC</td>
</tr>
</tbody>
</table>

Page 1 of 2

### Standard attributes

<table>
<thead>
<tr>
<th>Page</th>
<th>Description</th>
<th>Page counter</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE1</td>
<td>New page</td>
<td></td>
</tr>
<tr>
<td>PAGE2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Print attributes

- Resource name: 
- Print mode: 
- Output length: 01
- Uppercase: 

`..our people make the difference..`
Example

Windows

---

**Form: Display Windows: ZTEST11**

<table>
<thead>
<tr>
<th>Window</th>
<th>Meaning</th>
<th>Type</th>
<th>Default par.</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRAPH1</td>
<td>SATYAM LOGO</td>
<td>GRAPH</td>
<td></td>
</tr>
<tr>
<td>MAIN</td>
<td>Main window</td>
<td>MAIN</td>
<td>AS</td>
</tr>
<tr>
<td>WINDOW1</td>
<td>Header1</td>
<td>VAR</td>
<td>AS</td>
</tr>
<tr>
<td>WINDOW2</td>
<td>header 2</td>
<td>VAR</td>
<td>AS</td>
</tr>
</tbody>
</table>

Window 1 of 4

Standard attributes

<table>
<thead>
<tr>
<th>Window</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRAPH1</td>
<td>SATYAM LOGO</td>
</tr>
</tbody>
</table>

Window type: GRAPH
Default paragraph: 

---

Windows

..our people make the difference..
Formatting Options

Offset - specifying an offset of ‘n’, n left most characters are not displayed.
E.g    &symbol&   --->  1234576789
        &symbol+3&  ---->  456789

Output Length
   &symbol(3)&   ---->  123
   &symbol(7)&   ---->  1234567

Time Mask
   &Time&
   Normally 10:08:12
   (hh) : (mm): (ss)

/: Set Time Mask = ‘HH:MM’
   &Time& = 10:08

/: Set Time Mask = ‘HH hours MM minutes’
   &Time& = 10 hours 08 minutes
   &Time(Z)& = 10 hours 8 minutes
Example

**Date Mask**
/: Set Date Mask = ‘DDMMYY’
   &Date& = 01 05 01
/: Set Date Mask = ‘MMMM DD,YYYY’
   &Date& = May 01,2001

**Fill Character**
Leading spaces in a value can be replaced with a fill character. The ‘F’ option is used and character to be filled is specified.
E.g &KNA1-UMSAT& = 700.00
    &KNA1-UMSAT(F*)& = **700.00

**Space Compression**
&symbol(C)&
It has an effect of replacing each string of space characters with a single space and shifting the words to left as necessary to close up gaps. Also Leading spaces are completely removed.

**Omitting Leading Zeros**
&symbol(Z)&
&Day& = 01
&DAY(Z)& = 1