

Handy HTML Pop up Viewer in ABAP



Applies to:

Should work from the SAP Version 4.6C, Tested in SAP 4.6D Mini Basis System. For more information, visit the [ABAP homepage](#).

Summary

Whenever we are using HTML to format output for display in ABAP, like formatted Mail content to be sent to an external Mail ID, Testing becomes painful process as until and unless the mail sent out we cannot see the output in HTML. In such scenarios the HTML viewer will come handy to test the Format in HTML.

Author: Ravi Andela

Company: Enteg Infotech Private Limited, Bangalore, India

Created on: November 20, 2010

Author Bio



Ravi Andela, Working as ABAP Consultant in Enteg Infotech Private Limited.

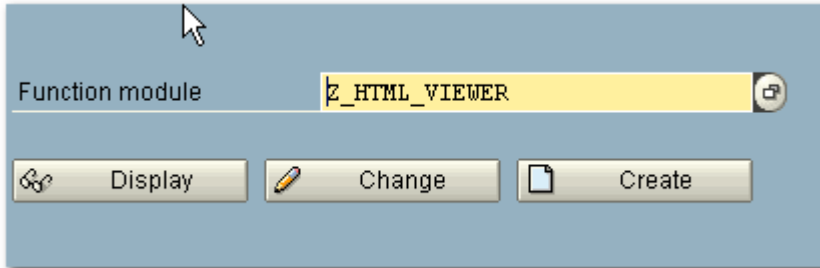
Table of Contents

Function Module for HTML Pop Up	3
Import Parameters	3
Source Code	3
Screen Attributes.....	4
Screen Flow Logic.....	4
Screen Layout	5
PF Status for the screen	6
PBO and PAI Modules of the Screen.....	6
Source Code for PBO and PAI	7
Global Data for Function Module	8
Test Program ZRAVI_HTML.....	9
Out Put in HTML Pop Up	11
Related Content.....	12
Disclaimer and Liability Notice.....	13

Function Module for HTML Pop Up

Create a Function Module for the pop up dialog screen and respective logic. This has the following import parameters.

- IM_HTML – for the HTML Data Table can be a text table of any length, can even pass string.
- IM_TITLE – for the title of the Pop Up Screen



Import Parameters

Parameter name	Type...	Reference type	Default value	Opt...	Pa...	Short text	Lo...
IM_HTML	TYPE	W3HTMLTAB		<input type="checkbox"/>	<input type="checkbox"/>	Table for ABAP Keyword Documentati...	...
IM_TITLE	TYPE	SY-TITLE		<input type="checkbox"/>	<input type="checkbox"/>	Screens, title text	...
				<input type="checkbox"/>	<input type="checkbox"/>		

Source Code

```

Function module: Z_HTML_VIEWER Active
Attributes Import Export Changing Tables Exceptions Source code

FUNCTION z_html_viewer.
  *-----
  ***Local interface:
  ** IMPORTING
  **   REFERENCE(IM_HTML) TYPE W3HTMLTAB
  **   REFERENCE(IM_TITLE) TYPE SY-TITLE
  *-----

  gt_html = im_html.
  gv_title = im_title.

  CALL SCREEN 100 STARTING AT 5 5.

ENDFUNCTION.

```

Screen Attributes

Screen Painter: Change Screen for SAPLZ_HTML

← → | | Layout

Screen number Active

Attributes | Element list | Flow logic

Short description

Original language English Development class

Last changed on/by

Last generation

Screen type

Normal

Subscreen

Modal dialog box

Selection screen

Settings

Hold data

Switch off runtime compress

Template - non-executable

Hold scroll position

Screen Flow Logic

Screen Painter: Change Screen for SAPLZ_HTML

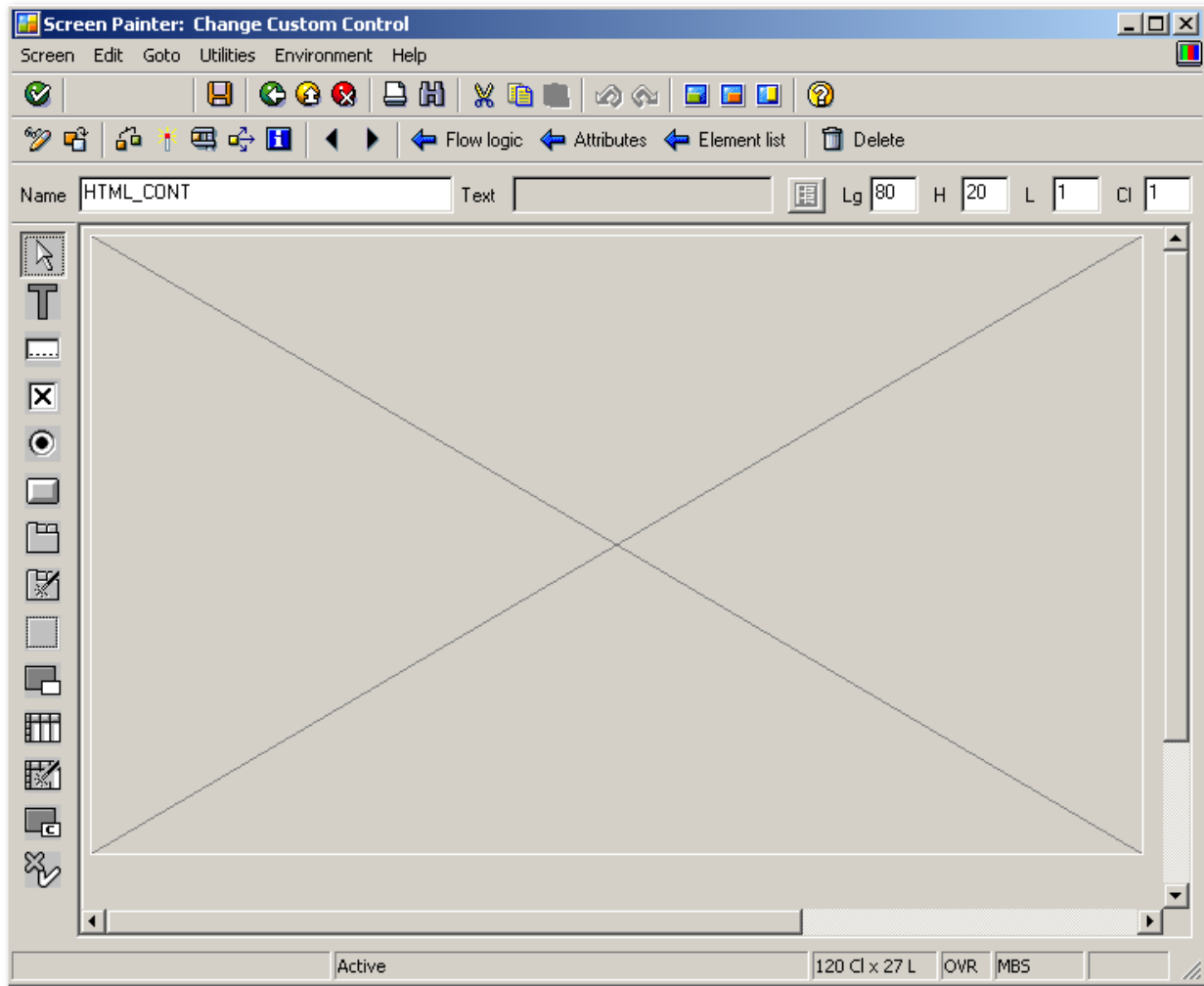
← → | | Layout | Pattern | Pretty Printer

Screen number Active

Attributes | Element list | Flow logic

```
PROCESS BEFORE OUTPUT.
  MODULE STATUS_0100.
*
PROCESS AFTER INPUT.
  MODULE USER_COMMAND_0100.
```

Screen Layout



PF Status for the screen

Display Status HTML, Interface SAPLZ_HTML

User interface **SAPLZ_HTML** Activated

Application toolbar **HTML Pop Up Screen**

Items 1 - 7	CONTINUE ✓	CANCEL ✗					
Items 8 - 14							
Items 15 - 21							
Items 22 - 28							
Items 29 - 35							

Function keys **HTML Pop Up Screen**

PBO and PAI Modules of the Screen

ABAP Editor: Change Include LZ_HTML001

Include **LZ_HTML001** Active

```

***INCLUDE LZ_HTML001 .
*-----*
*&-----*
*&      Module STATUS_0100  OUTPUT
*&-----*
*      text
*-----*
MODULE |status_0100 OUTPUT.
  SET PF-STATUS 'HTML'.
  SET TITLEBAR 'HTML_TITLE' WITH gv_title.

  DATA: lv_url TYPE char255.

  CREATE OBJECT gr_cont
    EXPORTING
      container_name = 'HTML_CONT'.

  CREATE OBJECT gr_html
    EXPORTING
      parent = gr_cont.

```

Source Code for PBO and PAI

```

*-----*
***INCLUDE LZ_HTML001 .
*-----*
*&-----*
*&      Module STATUS_0100  OUTPUT
*&-----*
*      text
*-----*
MODULE status_0100 OUTPUT.
  SET PF-STATUS 'HTML'.
  SET TITLEBAR 'HTML_TITLE' WITH gv_title.

  DATA: lv_url TYPE char255.

  CREATE OBJECT gr_cont
    EXPORTING
      container_name = 'HTML_CONT'.

  CREATE OBJECT gr_html
    EXPORTING
      parent = gr_cont.

  CALL METHOD gr_html->load_data
  * EXPORTING
  *   URL           =
  *   TYPE          = 'text'
  *   SUBTYPE       = 'html'
  *   SIZE          = 0
  IMPORTING
    assigned_url   = lv_url
  CHANGING
    data_table     = gt_html
  EXCEPTIONS
    dp_invalid_parameter = 1
    dp_error_general    = 2
    cntl_error          = 3
    OTHERS              = 4
    .
  IF sy-subrc <> 0.
  * MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
  *           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

  CALL METHOD gr_html->show_url
  EXPORTING
    url           = lv_url
  * FRAME        =
  * IN_PLACE     = ' X'
  EXCEPTIONS
    cntl_error          = 1
    cnht_error_not_allowed = 2

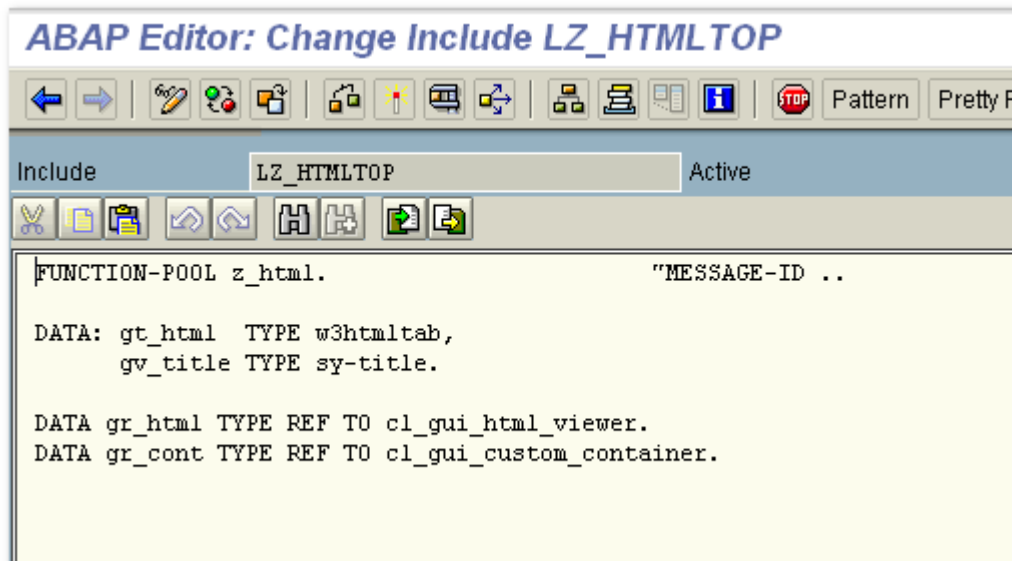
```

```

        cnht_error_parameter    = 3
        dp_error_general       = 4
        OTHERS                  = 5
    .
    IF sy-subrc <> 0.
    * MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
    *           WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
    ENDIF.
ENDMODULE.                " STATUS_0100  OUTPUT
*&-----*
*&      Module  USER_COMMAND_0100  INPUT
*&-----*
*           text
*-----*
MODULE user_command_0100 INPUT.
    CASE sy-ucomm.
        WHEN 'CONTINUE' OR 'CANCEL'.
            LEAVE TO SCREEN 0.
    ENDCASE.
ENDMODULE.                " USER_COMMAND_0100  INPUT

```

Global Data for Function Module



Test Program ZRAVI_HTML

REPORT zravi_html .

DATA: html TYPE w3htmltab.

DATA: lf_html TYPE string.

```

CONCATENATE '<html>'
  '<head>'
  '***Start of Styles .
  '<style type="text/css">'
  'h1 {'
  'color:#EFAB00;'
  'font-family:tahoma,Helvetica,sans-serif;'
  'font-size:160%;'
  'font-weight:bold;'
  'margin-bottom:8px;'
  'margin-top:8px;'
  'text-decoration:none;'
  'text-transform:uppercase;'
  '}'
  'h2 {'
  'color:#666666;'
  'font-family:tahoma,Helvetica,sans-serif;'
  'font-size:140%;'
  'font-weight:normal;'
  'margin-bottom:15px;'
  'margin-top:10px;'
  'text-decoration:none;'
  'text-transform:uppercase;'
  '}'
  'h3, h4, h5, h6 {'
  'color:#44697D;'
  'font-family:verdana,Helvetica,sans-serif;'
  'font-size:110%;'
  'font-weight:bold;'
  'margin-bottom:3px;'
  'text-decoration:none;'
  '}'
  'p {'
  'color:#333333;'
  'font-family:tahoma,Helvetica,sans-serif;'
  'margin-bottom:4px;'
  'margin-top:4px;'
  '}'
  'strong, b {'
  'font-family:tahoma,Helvetica,sans-serif;'
  'font-weight:bold;'
  '}'
  '</style>'
  '</head>'
  INTO lf_html SEPARATED BY space.

```

*** End of Styles

```

CONCATENATE lf_html
  '<body>'
  '<br>'
  '<h1> Heading Style 1 </h1>'
  '<br>'

```

```
'<h2> Heading Style 2 </h2>'
'<br>'
'<p>This is a paragraph example to be shown as a paragraph'
' Paragraph Line Number 2 this also displays the template '
'to be rendered at the run time and even this data can '
' be supplied at the runtime from database, but this data '
'is shown by hard coding in the program and even the '
'styles are hard coded inside the program. </p>'

'<h3> Check out the pre-requisites: </h3>'
'<p>Although style information may be set for almost'
' every HTML element, '
'two elements, DIV and SPAN, are particularly'
' useful in that they do not '
'impose any presentation semantics'
'(besides block-level vs. inline). When '
'combined with style sheets, these elements allow users to extend '
'<b>HTML</b> indefinitely, particularly when '
'used with the class and id attributes. </P>'
'<h3> IF you <strong> don''t fulfil </strong> '
'the pre-requisites: </h3>'

'<p>Although style information may be set for almost'
' every HTML element, '
'two elements, DIV and SPAN, are particularly'
' useful in that they do not '
'impose any presentation semantics'
'(besides block-level vs. inline). When '
'combined with style sheets, these elements allow users to extend '
'<b>HTML</b> indefinitely, particularly when '
'used with the class and id attributes. </P>'

'</body>'
'</html>'
```

INTO lf_html SEPARATED BY space.

```
DATA subrc TYPE c VALUE 'X'.
DATA len TYPE i.
DATA line TYPE char255.
```

```
WHILE subrc = 'X'.
```

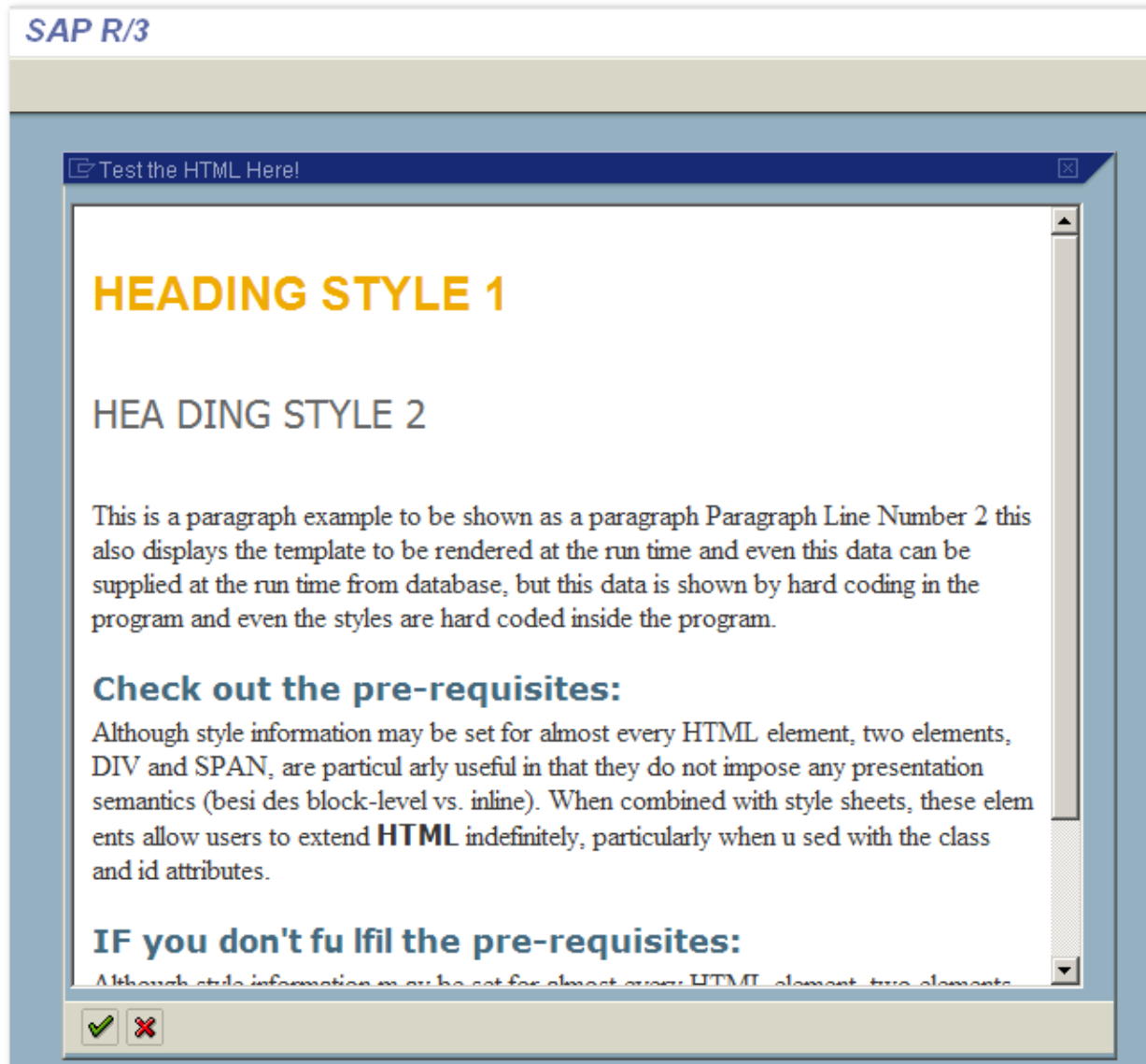
```
  len = strlen( lf_html ).
  IF len > 72.
    line = lf_html+0(72).
    APPEND line TO html.
    SHIFT lf_html BY 72 PLACES.
  ELSE.
```

```
    line = lf_html.
    APPEND line TO html.
    CLEAR subrc.
  ENDIF.
```

```
  CLEAR line.
ENDWHILE.
```

```
CALL FUNCTION 'Z_HTML_VIEWER'
  EXPORTING
    im_html = html
    im_title = 'Test the HTML Here!'.
```

Out Put in HTML Pop Up



Related Content

[Sap Controls Workshop an SAP Press Book](#)

[SDN Wiki Page for Java Script and HTML in ABAP](#)

[SDN The most useful Reference site for all the SAP Consultants](#)

For more information, visit the [ABAP homepage](#).

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.