

```

*&-----*
*& Report  ZALV_STYLES                                *
*&                                                *
*&-----*
*& AS : ALV Styles in a fieldcatalogue.                *
*& The ALV is developed using OOPS and various styles are given for *
*& each column in the fieldcatalogue                    *
*&-----*

```

```
REPORT zalv_styles.
```

```
*Include for ALV styles
INCLUDE <cl_alv_control>.
```

```
*Type ppols for alv
TYPE-POOLS : slis.
```

```
*structure for t582a tbale
TYPES : BEGIN OF ty_table,
        infty TYPE infty,
        pnnnn TYPE pnnnn_d,
        zrmkz TYPE dzrmkz,
        zeitb TYPE dzeitb,
        dname TYPE dianm,
        edynr TYPE edynp,
        ldynr TYPE ldynp,
        stypt TYPE stypt,
        sytxt TYPE sytxt,
        davo TYPE davo,
        davoe TYPE davoe,
        END OF ty_table.
```

```
*Structure for infotype text
TYPES : BEGIN OF ty_itext,
        infty TYPE infty,
        itext TYPE intxt,
        sprsl TYPE sprsl,
        END OF ty_itext.
```

```
*Structure for output display
TYPES : BEGIN OF ty_output,
        infty TYPE infty,
        itext TYPE intxt,
        pnnnn TYPE pnnnn_d,
        zrmkz TYPE dzrmkz,
        zeitb TYPE dzeitb,
        dname TYPE dianm,
        edynr TYPE edynp,
        ldynr TYPE ldynp,
        stypt TYPE stypt,
        sytxt TYPE sytxt,
        davo TYPE davo,
        davoe TYPE davoe,
        text(6) TYPE c,
        sradio(6) TYPE c,
        scheck(6) TYPE c,
        END OF ty_output.
```

```
*internal table and work area declarations
DATA : it_table TYPE STANDARD TABLE OF ty_table INITIAL SIZE 0,
        it_output TYPE STANDARD TABLE OF ty_output INITIAL SIZE 0,
        it_ittext TYPE STANDARD TABLE OF ty_itext INITIAL SIZE 0,
        wa_table TYPE ty_table,
```

```

        wa_output TYPE ty_output,
        wa_ittext TYPE ty_ittext.

*Data declarations for ALV
DATA: c_ccont TYPE REF TO cl_gui_custom_container, "Custom container object
      c_alvgd   TYPE REF TO cl_gui_alv_grid,      "ALV grid object
      it_fcat   TYPE lvc_t_fcat,                  "Field catalogue
      it_layout TYPE lvc_s_layo.                  "Layout

*Field symbols declarations for style

FIELD-SYMBOLS : <wa_fcat> TYPE lvc_s_fcat.

*initialization event

INITIALIZATION.

*start of selection event
START-OF-SELECTION.

*select the infotypes maintained
SELECT infty
      pnnnn
      zrmkz
      zeitb
      dname
      edynr
      ldynr
      stypt
      sytxt
      davo
      davoe
FROM t582a UP TO 25 ROWS
INTO CORRESPONDING FIELDS OF TABLE it_table.

* *Select the infotype texts
IF it_table[] IS NOT INITIAL.
  SELECT itext
        infty
        sprsl
        FROM t582s
        INTO CORRESPONDING FIELDS OF TABLE it_ittext
        FOR ALL ENTRIES IN it_table
        WHERE infty = it_table-infty
        AND sprsl = 'E'.
ENDIF.

*Appending the data to the internal table of ALV output
LOOP AT it_table INTO wa_table.

  wa_output-infty = wa_table-infty.
  wa_output-pnnnn = wa_table-pnnnn.
  wa_output-zrmkz = wa_table-zrmkz.
  wa_output-zeitb = wa_table-zeitb.
  wa_output-dname = wa_table-dname.
  wa_output-edynr = wa_table-edynr.
  wa_output-ldynr = wa_table-ldynr.
  wa_output-stypt = wa_table-stypt.
  wa_output-sytxt = wa_table-sytxt.
  wa_output-davo = wa_table-davo.
  wa_output-davoe = wa_table-davoe.

```

```

* For texts

  READ TABLE it_ittext INTO wa_ittext WITH KEY infty = wa_table-infty.
  wa_output-itext = wa_ittext-itext.
  wa_output-text = wa_ittext-sprsl.

  APPEND wa_output TO it_output.
  CLEAR wa_output.

ENDLOOP.

* Calling the ALV screen with custom container

  CALL SCREEN 0600.

*On this statement double click it takes you to the screen painter SE51.*Enter
the attributes
*Create a Custom container and name it CC_CONT and OK code as OK_CODE.
*Save check and Activate the screen painter.
*Now a normal screen with number 600 is created which holds the ALV grid.* PBO
of the actual screen ,
* Here we can give a title and customized menus
*&-----*
*&      Module  STATUS_0600  OUTPUT
*&-----*
*      text
*-----*
MODULE status_0600 OUTPUT.
*  SET PF-STATUS 'xxxxxxxx'.
*  SET TITLEBAR 'xxx'.
ENDMODULE.          " STATUS_0600  OUTPUT

* calling the PBO module ALV_GRID.
*&-----*
*&      Module  ALV_GRID  OUTPUT
*&-----*
*      text
*-----*
MODULE alv_grid OUTPUT.
  CREATE OBJECT c_ccont
    EXPORTING
      container_name = 'CC_CONT'.
  CREATE OBJECT c_alvgd
EXPORTING
i_parent = c_ccont.

*  SET field for ALV
  PERFORM alv_build_fieldcat.

* Setting the styles for the ALV grid control
* using field-symbols

  LOOP AT it_fcat ASSIGNING <wa_fcat>.

*For Each and every line of the fieldcat
  CASE sy-tabix.

*Color Styles
*Background/Font/Group/positive/negative
  WHEN '1'.
    <wa_fcat>-style = alv_style_color_inv_positive.
  WHEN '2'.
    <wa_fcat>-style = alv_style_color_int_negative.

```

```

    WHEN '3'.
      <wa_fcat>-style = alv_style_color_inv_negative.
    WHEN '4'.
      <wa_fcat>-style = alv_style_color_int_positive.
    WHEN '5'.
      <wa_fcat>-style = alv_style_color_background.
      <wa_fcat>-style = alv_style_color_inv_background.
    WHEN '6'.
      <wa_fcat>-style = alv_style_color_group.
      <wa_fcat>-style = alv_style_color_int_background.

*Style for F4
  WHEN '7'.
    <wa_fcat>-style = alv_style_f4.

*Style for Alignment(others are also possible)
  WHEN '8'.
    <wa_fcat>-style = alv_style_align_left_bottom.

*Style for Font Underlined/Bold and Italic are possible
  WHEN '9'.
    <wa_fcat>-style = alv_style_font_underlined.

*Style for button type
  WHEN '10'.
    <wa_fcat>-style = alv_style_button.

*Style for Font Symbol
  WHEN '11'.
    <wa_fcat>-style = alv_style_font_symbol.

*Style for Radiobutton
  WHEN '12'.
    <wa_fcat>-style = alv_style_radio_checked.

*Style for checkbox
  WHEN '13'.
    <wa_fcat>-style = alv_style_checkbox_checked.

*Style for column style characteristics(highlighting the col)
  WHEN '14'.
    <wa_fcat>-style = alv_col_style_characteristic.

*Styles for Enabling the column
  WHEN '15'.
    <wa_fcat>-style = alv_style_enabled.

  ENDCASE.

  ENDLLOOP.

* Set ALV attributes FOR LAYOUT
  PERFORM alv_report_layout.

  CHECK NOT c_alvkd IS INITIAL.

* Call ALV GRID

  CALL METHOD c_alvkd->set_table_for_first_display
    EXPORTING
      is_layout                = it_layout
    CHANGING
      it_outtab                = it_output

```

```

        it_fieldcatalog          = it_fcat
    EXCEPTIONS
        invalid_parameter_combination = 1
        program_error                 = 2
        too_many_lines                 = 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.

ENDMODULE.                " ALV_GRID  OUTPUT

*&-----*
*&      Form  alv_build_fieldcat
*&-----*
*          text
*-----*
*      <--P_IT_FCAT  text
*-----*
*subroutine to build fieldcat

FORM alv_build_fieldcat.

    DATA lv_fldcat TYPE lvc_s_fcat.
    CLEAR lv_fldcat.

    lv_fldcat-row_pos    = '1'.
    lv_fldcat-col_pos    = '1'.
    lv_fldcat-fieldname  = 'INFTY'.
    lv_fldcat-tabname    = 'IT_OUTPUT'.
    lv_fldcat-outputlen  = 8.
    lv_fldcat-scrtext_m  = 'Infotype'.
    lv_fldcat-icon       = 'X'.
    APPEND lv_fldcat TO it_fcat.
    CLEAR lv_fldcat.

    lv_fldcat-row_pos    = '1'.
    lv_fldcat-col_pos    = '2'.
    lv_fldcat-fieldname  = 'PNNNN'.
    lv_fldcat-tabname    = 'IT_OUTPUT'.
    lv_fldcat-outputlen  = 15.
    lv_fldcat-scrtext_m  = 'Structure'.
    lv_fldcat-icon       = ''.
    APPEND lv_fldcat TO it_fcat.
    CLEAR lv_fldcat.

    lv_fldcat-row_pos    = '1'.
    lv_fldcat-col_pos    = '3'.
    lv_fldcat-fieldname  = 'ITEXT'.
    lv_fldcat-tabname    = 'IT_OUTPUT'.
    lv_fldcat-outputlen  = 60.
    lv_fldcat-scrtext_m  = 'Description'.
    lv_fldcat-icon       = ''.
    APPEND lv_fldcat TO it_fcat.
    CLEAR lv_fldcat.

    lv_fldcat-row_pos    = '1'.
    lv_fldcat-col_pos    = '4'.
    lv_fldcat-fieldname  = 'TEXT'.
    lv_fldcat-tabname    = 'IT_OUTPUT'.
    lv_fldcat-outputlen  = 5.

```

```
lv_fldcat-scrtext_m = 'General'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos = '1'.
lv_fldcat-col_pos = '5'.
lv_fldcat-fieldname = 'ZRMKZ'.
lv_fldcat-tabname = 'IT_OUTPUT'.
lv_fldcat-outputlen = 1.
lv_fldcat-scrtext_m = 'PERIOD'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos = '1'.
lv_fldcat-col_pos = '6'.
lv_fldcat-fieldname = 'ZEITB'.
lv_fldcat-tabname = 'IT_OUTPUT'.
lv_fldcat-outputlen = 60.
lv_fldcat-scrtext_m = 'Time constraint'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos = '1'.
lv_fldcat-col_pos = '7'.
lv_fldcat-fieldname = 'DNAME'.
lv_fldcat-tabname = 'IT_OUTPUT'.
lv_fldcat-outputlen = 15.
lv_fldcat-scrtext_m = 'Dialogmodule'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos = '1'.
lv_fldcat-col_pos = '8'.
lv_fldcat-fieldname = 'EDYNR'.
lv_fldcat-tabname = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'Single screenno'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos = '1'.
lv_fldcat-col_pos = '9'.
lv_fldcat-fieldname = 'LDYNR'.
lv_fldcat-tabname = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'List screenno'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos = '1'.
lv_fldcat-col_pos = '10'.
lv_fldcat-fieldname = 'STYPT'.
lv_fldcat-tabname = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
```

```
lv_fldcat-scrtext_m = 'SubtypeTable'.
lv_fldcat-icon = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos   = '1'.
lv_fldcat-col_pos   = '11'.
lv_fldcat-fieldname = 'SYTXT'.
lv_fldcat-tabname   = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'Font Symbol'.
lv_fldcat-icon      = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos   = '1'.
lv_fldcat-col_pos   = '12'.
lv_fldcat-fieldname = 'SRADIO'.
lv_fldcat-tabname   = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'RADIO'.
lv_fldcat-icon      = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos   = '1'.
lv_fldcat-col_pos   = '13'.
lv_fldcat-fieldname = 'SCHECK'.
lv_fldcat-tabname   = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'CHECK'.
lv_fldcat-icon      = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos   = '1'.
lv_fldcat-col_pos   = '14'.
lv_fldcat-fieldname = 'DAVO'.
lv_fldcat-tabname   = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'Start Date'.
lv_fldcat-icon      = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
lv_fldcat-row_pos   = '1'.
lv_fldcat-col_pos   = '15'.
lv_fldcat-fieldname = 'DAVOE'.
lv_fldcat-tabname   = 'IT_OUTPUT'.
lv_fldcat-outputlen = 10.
lv_fldcat-scrtext_m = 'End date'.
lv_fldcat-icon      = ''.
APPEND lv_fldcat TO it_fcat.
CLEAR lv_fldcat.
```

```
ENDFORM.                " alv_build_fieldcat
*&-----*
*&      Form  alv_report_layout
*&-----*
*      text
*-----*
*      <--P_IT_LAYOUT  text
```

```

*-----*
*Subroutine for setting alv layout
FORM alv_report_layout.
  it_layout-cwidth_opt = 'X'.
  it_layout-zebra = 'X'.
  it_layout-col_opt = 'X'.

ENDFORM.                " alv_report_layout
* PAI module of the screen created. In case we use an interactive ALV or
*for additional functionalities we can create OK codes
*and based on the user command we can do the coding.
*&-----*
*&      Module  USER_COMMAND_0600  INPUT
*&-----*
*      text
*-----*
MODULE user_command_0600 INPUT.

ENDMODULE.                " USER_COMMAND_0600  INPUT

```

Output
Part 1

Infotype	Structure	Description	General	PERIOD	Time constraint	Dialogmodule	Single screenno	List screenno	SubtypeTable
0000	P0000	0000 Actions	E		1	RP_0000	2000	3000	
0001	P0001	0001 Organizational Assignment	E		1	RP_0001	2000	3000	
0002	P0002	0002 Personal Data	E		1	RP_0002	2000	3000	
0003	P0003	0003 Payroll Status	E		A	RP_0003	2000	3000	
0004	P0004	0004 Challenge	E		2	RP_0004	2000	3000	
0005	P0005	0005 Leave Entitlement	E		2	RP_0005	2000	3000	JAHR

(Continued output)
Part 2

General	PERIOD	Time constraint	Dialogmodule	Single screenno	List screenno	SubtypeTable	Font Symbol	RADIO	CHECK	Start Date	End date
E		1	RP_0000	2000	3000			<input type="radio"/>	<input type="checkbox"/>		1
E		1	RP_0001	2000	3000			<input type="radio"/>	<input type="checkbox"/>	1	1
E		1	RP_0002	2000	3000			<input type="radio"/>	<input type="checkbox"/>	3	1