

CODE SAMPLE – ALV GRID

The code submitted below enables the display of ALV GRID with additional options such as:

- 1) Coloring of individual cells.
- 2) Making the cell contents invisible.
- 3) Layout selection at the selection screen level before displaying the grid.

Summary:

- 1) The subroutine **PERFORM INVISIBLE_CELL_CONTENTS** at page 4 – line 38 of this text document performs the necessary routine to make the cell contents invisible.
- 2) The subroutines **INIT_VARIANT, VARIANT_SELECT, VARIANT_FILL** at lines 54 to 59 in page 2 of this document Performs the necessary routine to select a previously saved Layouts of the ALV GRID.

I have attached the screen shots of the above mentioned operations in the ALV GRID. Please do review the content and provide feedback.

```
report zc9_alvgrid_final_now .
*-----
*PROGRAM ON ALV-GRID.
*-----
*-----
*TABLE DECLARATION.
*-----
*tables:zc9_employee, zc9_empspec.
type-pools: slis.                                "ALV Declarations
data: celcol1 type slis_specialcol_alv.
data: syprindex type i value 1.
*-----
*INTERNAL TABLE DECLARATION.
*-----

data: begin of itab_alv1 OCCURS 0,
      emp_id(5) TYPE C,
      emp_name(20) TYPE C,
      certified(3) TYPE C,
      celcol type slis_t_specialcol_alv,
      selected(1) type c,
end of itab_alv1.

data: begin of itab_alv3 OCCURS 0,
      emp_id(5) TYPE C,
      emp_name(20) TYPE C,
      certified(3) TYPE C,
end of itab_alv3.

data: mulrowtab1 like line of itab_alv1.
data: mulrowtab like table of itab_alv1.
data: wa_emp_name like zc9_employee-emp_name.
```

```

data: mulrowindex type i value 1.
data: maxlines type i.
*-----
data: wa like line of itab_alv1.
data: wa_itab like line of itab_alv1.
data: ok-code like sy-ucomm.
data: ok-code1 like sy-ucomm.
data: ind type i.
data: newvar type disvariant.
data: tempvar type i value 0.
data: gd_repid like sy-repid.
*-----
*ALV-GRID DATA DECLARATION.
*-----
data: fieldcatalog type slis_t_fieldcat_alv.
*                               FIELD CATALOG.
data: fieldcatalog1 type slis_fieldcat_alv.
data: gd_layout type slis_layout_alv.
*                               LAYOUT DECLARATION.
data: id_color type c.
data: eventtable type slis_t_event.
data: wa_event type slis_alv_event.
data: newvariant type disvariant.
data: i_exclude type slis_t_extab,           "ALV Exclusion Table
      wa_exclude type slis_extab.           "ALV Exclusion Structure

*-----
*VARIANT DECLARATION.
*-----
data: vartype like disvariant.
data: var_save(1) type c.
data: w_exit(1) type c.
data: wx_variant like disvariant.
data: variant_para like disvariant-variant.
data: itab_emp like table of zc9_employee.
data: itab_emp1 like line of itab_emp.
data: itab_spec like table of zc9_empspec.
data: itab_spec1 like line of itab_spec.
data: ls_sort type slis_sortinfo_alv.
data: lt_sort type slis_t_sortinfo_alv.
parameter: LAYOUT like disvariant-variant.
*-----
*  INITIALIZATION.
*-----
gd_repid = sy-repid.
*-----
*CALLING SELECTION SCREEN-SUBROUTINES.
*-----
*-----
*AT SELECTION SCREEN
*-----
at selection-screen output.
  perform init_variant.
  perform variant_default using LAYOUT.

at selection-screen on value-request for LAYOUT.
  perform variant_select using LAYOUT.

at selection-screen.
  perform variant_fill.
  newvar = vartype.
*-----

```

* LAYOUT DECLARATION.

```
* -----
gd_layout-zebra = 'X'.
gd_layout-no_input      = 'X'.
gd_layout-colwidth_optimize = 'X'.
gd_layout-box_fieldname = 'SELECTED'.
gd_layout-box_tabname   = 'ITAB_ALV1'.
gd_layout-totals_text   = 'Totals'(201).
gd_layout-coltab_fieldname = 'CELCOL'.
gd_layout-f2code        = 'DISP'. "Sets fcode for when double
*                               "click(press f2)
```

```
call function 'REUSE_ALV_FIELDATALOG_MERGE'
exporting
  i_program_name      = gd_repid
  i_internal_tabname  = 'ITAB_ALV3'
  i_inclname          = gd_repid
changing
  ct_fieldcat         = fieldcatalog
.
```

* -----
* FIELD-CATALOG DECLARATION.

```
* -----
loop at fieldcatalog into fieldcatalog1.
  case fieldcatalog1-fieldname.

    when 'EMP_ID'.
      fieldcatalog1-seltext_m = 'EMP-ID'.
      fieldcatalog1-col_pos   = 0.
      fieldcatalog1-outputlen = 10.
      fieldcatalog1-emphasize = 'C2'.
      fieldcatalog1-key       = 'X'.
      fieldcatalog1-do_sum    = 'X'.
      fieldcatalog1-sp_group  = 'A'.

    when 'EMP_NAME'.
      fieldcatalog1-fieldname = 'EMP_NAME'.
      fieldcatalog1-seltext_m = 'EMPLOYEE_NAME'.
      fieldcatalog1-col_pos   = 1.
      fieldcatalog1-emphasize = 'C2'.
      fieldcatalog1-hotspot   = 'X'.
      fieldcatalog1-key       = 'X'.
      fieldcatalog1-outputlen = 20.
      fieldcatalog1-sp_group  = 'B'.

    when 'CERTIFIED'.
      fieldcatalog1-fieldname = 'CERTIFIED'.
      fieldcatalog1-seltext_m = 'CERTIFIED'.
      fieldcatalog1-col_pos   = 2.
      fieldcatalog1-emphasize = 'C2'.
      fieldcatalog1-key       = 'X'.
      fieldcatalog1-sp_group  = 'C'.
  endcase.
  modify fieldcatalog from fieldcatalog1.
endloop.
```

* -----
* SORTING FIELD CATALOG FIELD.

```
* -----
ls_sort-spos = 1.
ls_sort-fieldname = 'EMP_ID'.
```

```
ls_sort-up = 'X'.
ls_sort-subtot = 'X'.
append ls_sort to lt_sort.
```

```
*-----
*POPULATING THE INTERNAL TABLE.
*-----
```

```
ITAB_ALV1-EMP_ID = '15530'.
ITAB_ALV1-EMP_NAME = 'THILAKJI'.
ITAB_ALV1-CERTIFIED = 'YES'.
APPEND ITAB_ALV1.
```

```
ITAB_ALV1-EMP_ID = '15531'.
ITAB_ALV1-EMP_NAME = 'MATHEWS'.
ITAB_ALV1-CERTIFIED = 'NO'.
APPEND ITAB_ALV1.
```

```
ITAB_ALV1-EMP_ID = '15532'.
ITAB_ALV1-EMP_NAME = 'JAISI'.
ITAB_ALV1-CERTIFIED = 'YES'.
APPEND ITAB_ALV1.
```

```
ITAB_ALV1-EMP_ID = '15533'.
ITAB_ALV1-EMP_NAME = 'ADAMS'.
ITAB_ALV1-CERTIFIED = 'YES'.
APPEND ITAB_ALV1.
```

```
ITAB_ALV1-EMP_ID = '15534'.
ITAB_ALV1-EMP_NAME = 'PAUL'.
ITAB_ALV1-CERTIFIED = 'NO'.
APPEND ITAB_ALV1.
```

```
*****
*
* FORM TO MAKE THE CELL CONTENTS INVISIBLE.
```

PERFORM INVISIBLE_CELL_CONTENTS.

```
*
*****
*-----
*EXCLUDE-DECLARATION.
*-----
```

```
clear wa_exclude.
wa_exclude-fcode = '&VEXCEL'.
append wa_exclude to i_exclude.
```

```
*-----
* FUNCTION MODULE CALL-REUSE-ALV-GRID-DISPLAY.
*-----
```

```
call function 'REUSE_ALV_GRID_DISPLAY'
  exporting
    i_callback_program      = gd_repid
    i_callback_top_of_page  = 'TOP-OF-PAGE2' "see FORM
    i_callback_user_command = 'USER_COMMAND'
    i_grid_title            = 'ALV GRID-EMPLOYEE'

    is_layout              = gd_layout
    it_fieldcat            = fieldcatalog[]
    it_sort                = lt_sort
    it_excluding           = i_exclude
    i_callback_pf_status_set = 'SET_PF_STATUS' "see FORM
    i_save                 = 'A'
```

```

        is_variant          = newvar
tables
        t_outtab            = itab_alv1
exceptions
        program_error       = 1
        others              = 2.

* -----*
* Form TOP-OF-PAGE*
* -----*
* ALV Report Header*
* -----*
form top-of-page2.
    data: header type slis_t_listheader,
          wa type slis_listheader,
          infield like wa-info,
          nline type i,
          nlinechar(2) type c.

*TITLE AREA
    wa-typ = 'H'.
    wa-info = 'EMPLOYEE-DETAILS'.
    append wa to header.

*BELOW AREA.
    wa-typ = 'S'.
    wa-key = 'DATE'.
concatenate sy-datum+6(2) '.' sy-datum+4(2) '.' sy-datum(4) into wa-info
.
    append wa to header.

*NEXT LINE.
    wa-typ = 'S'.
    describe table itab_alv1 lines nline.
    nlinechar = nline.
    wa-key = 'TABLE DETAILS'.
    concatenate 'NO.of.Lines:' nlinechar into wa-info separated by space.
    append wa to header.

    call function 'REUSE_ALV_COMMENTARY_WRITE'
        exporting
            it_list_commentary = header
            i_logo              = 'ALVLOGO'.

endform.

* -----*
* FORM SET_PF_STATUS*
* -----*
form set_pf_status using rt_extab type slis_t_extab.
    set pf-status 'NEWSTATUS123'.

endform.

* -----*
* FORM USER_COMMAND*
* -----*
* --> R_UCOMM*
* --> RS_SELFIELD*
* -----*
form user_command using r_ucomm like sy-ucomm
    rs_selfield type slis_selfield.

```

```

* Check function code
case r_ucomm.

    when 'SELM'.
        loop at itab_alv1 where selected = 'X'.
            move-corresponding itab_alv1 to mulrowtab1.
            append mulrowtab1 to mulrowtab.
        endloop.
        message i029(zmc).
        clear mulrowtab1.

    when 'DISP'.
        * Check field clicked on within ALVgrid report
        if rs_selfield-fieldname = 'EMP_ID'.
        * Read data table, using index of row user clicked on
            read table itab_alv1 into wa index rs_selfield-tabindex.
            ind = rs_selfield-tabindex.
            message i025(zmc) with rs_selfield-fieldname .

            elseif rs_selfield-fieldname = 'EMP_NAME'.
                read table itab_alv1 into wa index rs_selfield-tabindex.
                ind = rs_selfield-tabindex.
                message i028(zmc) with rs_selfield-fieldname wa-emp_name .
            endif.

    when 'EXIT'.
        * MESSAGE I026(ZMC).
        leave program.

    when 'CHAN'.
        * READ TABLE itab_alv1 INTO wa INDEX rs_selfield-tabindex.
        read table itab_alv1 into wa index rs_selfield-tabindex.
        call screen 1551.
        tempvar = 1.

* -----
* CALLING THE GRID DISPLAY FUNCTION AGAIN.
* -----
* FUNCTION MODULE CALL-REUSE-ALV-GRID-DISPLAY.
* -----

    call function 'REUSE_ALV_GRID_DISPLAY'
        exporting
            i_callback_program      = gd_repid
            i_callback_top_of_page  = 'TOP-OF-PAGE' "see FORM
            i_callback_user_command = 'USER_COMMAND'
            i_grid_title            = 'ALV GRID-EMPLOYEE'
            is_layout               = gd_layout
            it_fieldcat             = fieldcatalog[]
            it_events               = eventtable
            i_save                  = 'X'
            it_excluding            = i_exclude
            i_callback_pf_status_set = 'SET_PF_STATUS' "see FORM
            is_variant              = newvar
        tables
            t_outtab                = itab_alv1
        exceptions
            program_error           = 1
            others                  = 2.

```

```

        r_ucomm = '&F03'.
    when '&F03'.
* IF TEMPVAR = 1.
        leave program.
    endcase.
clear r_ucomm.
endform.
*&-----*
*&      Module  USER_COMMAND_1551  INPUT
*&-----*
*      text
*-----*
module user_command_1551 input.

    data: syprtabix type i.
    case ok-code.
        when 'QUIT'.
            set screen 0.
            leave screen.
            clear itab_alv1.
            sy-tabix = 0.

        when 'UPDA'.
            loop at itab_alv1 where emp_id = wa-emp_id.
                syprtabix = sy-tabix.
            endloop.

    modify itab_alv1 from wa index syprtabix transporting emp_id emp_name
        certified.

    message i027(zmc).

    leave screen.
endcase.
endmodule.                " USER_COMMAND_1551  INPUT
*&-----*
*&      Module  STATUS_1551  OUTPUT
*&-----*
*      text
*-----*
module status_1551 output.
    set pf-status 'ZNEWSTATUS'.
*   SET TITLEBAR 'xxx'.

endmodule.                " STATUS_1551  OUTPUT

*-----*
*FORM FOR SELECTING LAYOUT.
*-----*

form variant_select using form_varpar.

    call function 'LVC_VARIANT_F4'
        exporting
            is_variant                = vartype
*   IT_DEFAULT_FIELDCAT              =
            i_save                    = var_save
        importing
            e_exit                    = w_exit
            es_variant                = wx_variant
        exceptions

```

```

        not_found          = 1
        program_error      = 2
        others              = 3
    .
    if sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
*   WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
    endif.

    if w_exit is initial.
        vartype-variant = wx_variant-variant.
        form_varpar = wx_variant-variant.
    endif.
endform.

*&-----*
*&      Form  init_variant
*&-----*
form init_variant.

    clear: vartype.
    gd_repid          = sy-repid.
    vartype-report     = sy-repid.
    vartype-username   = sy-uname.
    var_save           = 'A'. "All types

endform.

*&-----*
*&      Form  variant_default
*&-----*
form variant_default using p_variant.

    wx_variant = vartype.

    if not p_variant is initial.
        wx_variant-variant = p_variant.
    endif.

    call function 'LVC_VARIANT_DEFAULT_GET'
        exporting
            i_save          = var_save
        changing
            cs_variant       = wx_variant
        exceptions
            wrong_input      = 1
            not_found        = 2
            program_error     = 3
            others            = 4.

    case sy-subrc.
        when 0.
            p_variant = wx_variant-variant.
        when 2.
            clear: p_variant.
    endcase.

endform.

*&-----*
*&      Form  variant_fill
*&-----*

```

```

form variant_fill.

clear: vartype.

if LAYOUT is initial.
    vartype-variant = 'STANDARD'.
    vartype-report   = sy-repid.
else.

    vartype-variant = LAYOUT.
    vartype-report   = sy-repid.

call function 'LVC_VARIANT_EXISTENCE_CHECK'
    exporting
        i_save      = var_save
    changing
        cs_variant = vartype
    exceptions
        others      = 01.
if sy-subrc ne 0.
    message i000(zz) with text-g02.
endif.
endif.
endform.

*&-----*
*&      Module  STATUS_1234  OUTPUT                *
*&-----*
*      text                                         *
*-----*
module status_1234 output.

*  SET PF-STATUS 'xxxxxxxx'.
*  SET TITLEBAR 'xxx'.

endmodule.                " STATUS_1234  OUTPUT
*&-----*
*&      Module  USER_COMMAND_1234  INPUT          *
*&-----*
*      text                                         *
*-----*
module user_command_1234 input.

sort mulrowtab.
describe table mulrowtab lines maxlines.

case ok-code1.
when 'OUT1'.
set screen 0.
leave screen.

when 'PUSH'.
clear mulrowtab1.
if mulrowindex <= maxlines.
read table mulrowtab into mulrowtab1 index mulrowindex.
mulrowindex = mulrowindex + 1.

else.
message i030(zmc).
endif.
endcase.
endmodule.                " USER COMMAND 1234  INPUT

```

```

* -----
* CEL COLOR TABLE POPULATION.
* -----
FORM INVISIBLE_CELL_CONTENTS.
  loop at itab_alv1 INTO WA.
    IF WA-EMP_ID = ' '.
      EXIT.
    ENDIF.
  clear itab_alv1.
  move-corresponding WA to itab_alv1.

  clear celcol1.
  celcol1-fieldname = 'EMP_ID'.
  celcol1-color-col = syprindex.
  append celcol1 to itab_alv1-celcol.

  celcol1-fieldname = 'EMP_NAME'.
  celcol1-color-col = syprindex + 2.
  append celcol1 to itab_alv1-celcol.

*****
* CODE FOR MAKING THE GRID CELL CONTENTS INVISIBLE BASED ON THE VALUE OF
* THE CELL. THE CONDITION CHECKED IS WHETHER
* CERTIFIED FIELD HAS VALUE 'NO'.
*****
  celcol1-fieldname = 'CERTIFIED'.
  if itab_alv1-certified = 'NO'.
    itab_alv1-emp_name = ' '.
  endif.
  celcol1-color-col = syprindex + 3.
  append celcol1 to itab_alv1-celcol.
  MODIFY ITAB_ALV1.
  syprindex = celcol1-color-col - 1.
  CLEAR ITAB_ALV1.
endloop.
ENDFORM.

```

SCREEN SHOTS

Description:

- ❖ This screen shot displays the ALV GRID with each cells having different color i.e., coloring of Individual ALV GRID cells.
- ❖ The cell contents are made invisible.

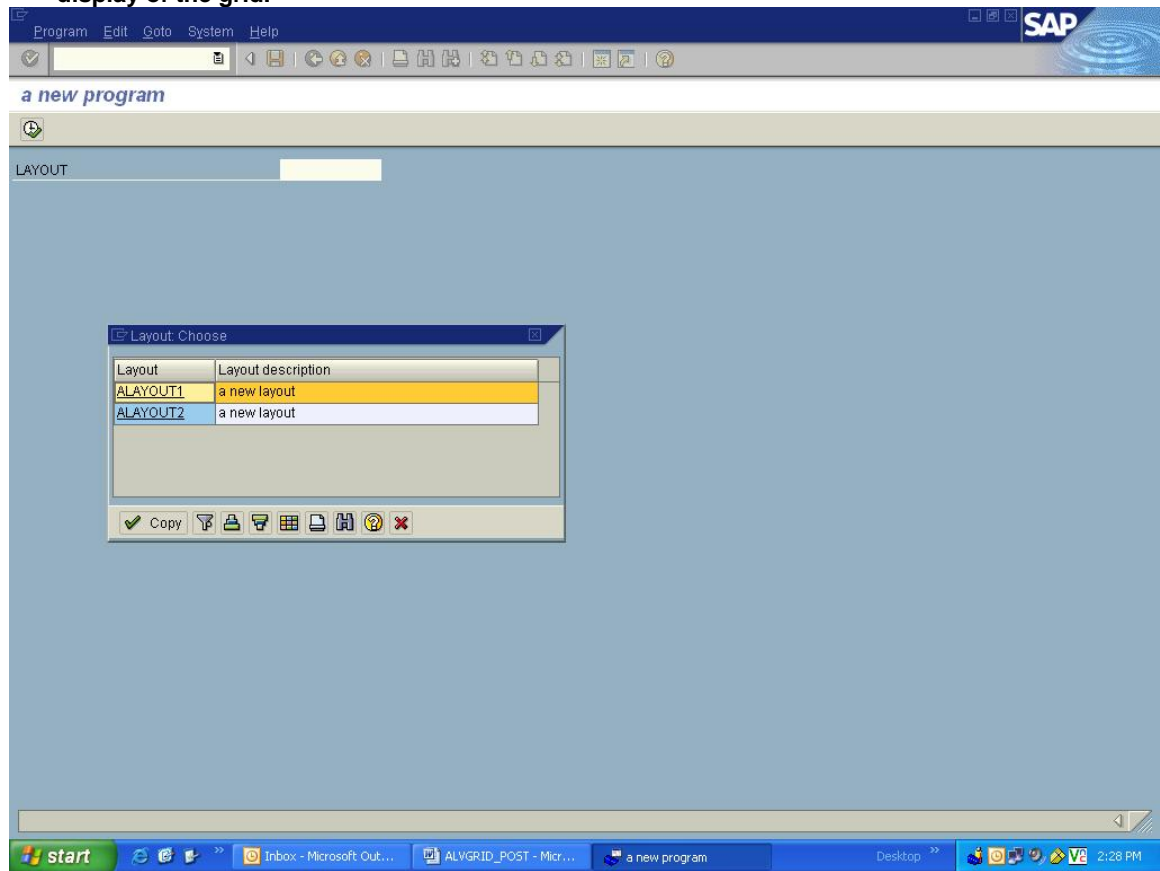
The screenshot shows the SAP ALV GRID-EMPLOYEE interface. The title bar indicates 'a new program'. The main window displays 'EMPLOYEE-DETAILS' with a date of 22.12.2004 and a table with 5 lines. The table is titled 'ALV GRID-EMPLOYEE' and contains the following data:

| EMP-ID | EMPLOYEE_NAME | CERTIFIED |
|--------|---------------|-----------|
| 15530 | THILAKJI | YES |
| 15531 | | NO |
| 15532 | JAJSI | YES |
| 15533 | ADAMS | YES |
| 15534 | | NO |

The cells in the table are colored differently (yellow, green, orange, blue, and grey) and the content is invisible.

Description:

- ❖ This screen shot displays the layout selection option that could be done before the display of the grid.



**Content submitted by,
Prabaharan Gnanasekaran,
Project – Engineer – Wipro Technologies,
Sholinganallur,
Chennai.**