

```

*&-----*
*& Report   ZALV_OOINTERACTIVE                               *
*& Author   Swarna.S.                                       *
*&-----*
*&       AS : Simple Interactive ALV report developed using OOPS *
*&                                               *
*&-----*

```

```
REPORT ZALV_OOINTERACTIVE.
```

```
*Class definition for handling double click
CLASS event_class DEFINITION DEFERRED.
```

```
*Internal table and work area declarations for dd02l and dd03l
```

```
DATA : it_dd02l TYPE TABLE OF dd02l,
      wa_dd02l TYPE dd02l,
      it_dd03l TYPE TABLE OF dd03l,
      wa_dd03l TYPE dd03l.
```

```
*data declarations for ALV Main list
```

```
DATA : ty_lay1 TYPE lvc_s_layo,
      it_fieldcat TYPE lvc_t_fcat ,
      ty_fieldcat TYPE lvc_s_fcat ,
      c_alv1 TYPE REF TO cl_gui_alv_grid,
      c_cont1 TYPE REF TO cl_gui_custom_container,
      event_receiver TYPE REF TO event_class.
```

```
*data declarations for ALV Interactive list
```

```
DATA : ty_lay2 TYPE lvc_s_layo,
      it_fcat TYPE lvc_t_fcat ,
      ty_fcat TYPE lvc_s_fcat ,
      c_alv2 TYPE REF TO cl_gui_alv_grid,
      c_cont2 TYPE REF TO cl_gui_custom_container.
```

```
**Select options for multiple values and NOT ranges
SELECT-OPTIONS : s_table FOR wa_dd02l-tabname NO INTERVALS.
```

```
* Initialization event
INITIALIZATION.
```

```
*Start of selection event
START-OF-SELECTION.
```

```
*fetch data into table and field characteristics
PERFORM fetch_data.
```

```
*ALV display for output
PERFORM alv_output.
```

```

*&-----*
*&       Form   FETCH_DATA
*&-----*
*       text
*-----*
* --> p1       text
* <-- p2       text
*-----*
FORM fetch_data .

```

```
*Selec the table details
```

```
SELECT * FROM dd021 INTO CORRESPONDING FIELDS OF TABLE it_dd021 WHERE tabname
IN s_table
AND tabclass = 'TRANSP'.
```

```
ENDFORM. " FETCH_DATA
```

```
*-----*
```

```
* CLASS lcl_event_receiver DEFINITION
```

```
*-----*
```

```
*
```

```
*-----*
```

```
CLASS event_class DEFINITION.
```

```
*Handling double click
PUBLIC SECTION.
```

```
METHODS:
```

```
handle_double_click
```

```
FOR EVENT double_click OF cl_gui_alv_grid IMPORTING e_row .
```

```
ENDCLASS. "lcl_event_receiver DEFINITION
```

```
*-----*
```

```
* CLASS lcl_event_receiver IMPLEMENTATION
```

```
*-----*
```

```
*
```

```
*-----*
```

```
CLASS event_class IMPLEMENTATION.
```

```
METHOD handle_double_click.
```

```
DATA : ls_dd021 LIKE LINE OF it_dd021.
```

```
*Reading the selected data into a variable
```

```
READ TABLE it_dd021 INDEX e_row-index INTO ls_dd021.
```

```
* *Select the field details of the selected table
```

```
SELECT * FROM dd031 INTO CORRESPONDING FIELDS OF TABLE it_dd031
WHERE tabname EQ ls_dd021-tabname.
```

```
*calling the ALV containing the field values
```

```
CALL SCREEN 101.
```

```
ENDMETHOD. "handle_double_click
```

```
ENDCLASS. "lcl_event_receiver IMPLEMENTATION
```

```
*&-----*
```

```
*& Module pbo_100 OUTPUT
```

```
*&-----*
```

```
MODULE pbo_100 OUTPUT.
```

```

*set pf-status 'XXX'.
*set titlebar 'XXX'.
ENDMODULE. " PBO_100 OUTPUT

*&-----*
*& Module alv_100 OUTPUT
*&-----*

MODULE alv_100 OUTPUT.

*Check if there is no custom container in screen 100
  IF c_cont1 IS INITIAL.

*Creating object of container
  CREATE OBJECT c_cont1
  EXPORTING
    container_name = 'CCONT'.
  IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

*Creating object of alv
  CREATE OBJECT c_alv1
  EXPORTING
    i_parent = c_cont1.
  IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

*alv layout
  PERFORM alv_100_layout.

*alv field catalogue
  PERFORM alv_100_fieldcat.

*Displaying the ALV grid
  CALL METHOD c_alv1->set_table_for_first_display
  EXPORTING
    is_layout      = ty_lay1
  CHANGING
    it_outtab      = it_dd021[]
    it_fieldcatalog = it_fieldcat.

  IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

*Create object of the event class and setting handler for double click
  CREATE OBJECT event_receiver.
  SET HANDLER event_receiver->handle_double_click FOR c_alv1.

  ENDIF.

ENDMODULE. " ALV_100 OUTPUT

*&-----*
*& Module pai_100 INPUT

```

```

*&-----*
MODULE pai_100 INPUT.
ENDMODULE. " pai_100 INPUT

*-----*

* MODULE PBO_101 OUTPUT

*-----*

MODULE pbo_101 OUTPUT.
* SET PF-STATUS 'XXX'.
* SET TITLEBAR 'XXX'.
ENDMODULE. " PBO_101 INPUT

*-----*

* MODULE ALV_101 OUTPUT

*-----*

MODULE alv_101 OUTPUT.

*Check if the Custom container exists.
  IF c_cont2 IS INITIAL.

*Creating container object
  CREATE OBJECT c_cont2
  EXPORTING
    container_name = 'CDCONT'.
  IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

*creating ALV grid for interactive list
  CREATE OBJECT c_alv2
  EXPORTING
    i_parent = c_cont2.
  IF sy-subrc <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

*ALV layout
  PERFORM alv_101_layout.

*ALV fieldcatalogue
  PERFORM alv_101_fieldcat.

*Sorting the output by field position
  SORT it_dd031 BY position.

*ALV for display field details
  CALL METHOD c_alv2->set_table_for_first_display
  EXPORTING
    is_layout      = ty_lay2
  CHANGING
    it_outtab      = it_dd031[]
    it_fieldcatalog = it_fcat.
  IF sy-subrc <> 0.

```

```

* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
  ENDIF.

ENDIF.

ENDMODULE. " ALV_101 OUTPUT

*&-----*
*& Module PAI_101 INPUT
*&-----*

MODULE pai_101 INPUT.

ENDMODULE. " PAI_101 INPUT
*&-----*
*&      Form ALV_OUTPUT
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM alv_output .
  CALL SCREEN 100.

ENDFORM.          " ALV_OUTPUT
*&-----*
*&      Form ALV_100_LAYOUT
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM alv_100_layout .

  ty_lay1-grid_title = 'TABLES'.
  ty_lay1-zebra = 'X'.
  ty_lay1-no_toolbar = 'X'.

ENDFORM.          " ALV_100_LAYOUT
*&-----*
*&      Form ALV_100_FIELDCAT
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM alv_100_fieldcat .

CLEAR ty_fieldcat.
ty_fieldcat-row_pos = 1.
ty_fieldcat-col_pos = 1.
ty_fieldcat-fieldname = 'TABNAME'.
ty_fieldcat-tabname = 'GT_DD02L'.
ty_fieldcat-coltext = 'TableName'.

```

```
ty_fieldcat-outputlen = 10.
APPEND ty_fieldcat TO it_fieldcat.
CLEAR ty_fieldcat.
```

```
ty_fieldcat-row_pos = 1.
ty_fieldcat-col_pos = 2.
ty_fieldcat-fieldname = 'TABCLASS'.
ty_fieldcat-tabname = 'GT_DD02L'.
ty_fieldcat-coltext = 'CATEGORY'.
ty_fieldcat-outputlen = 10.
APPEND ty_fieldcat TO it_fieldcat.
CLEAR ty_fieldcat.
```

```
ty_fieldcat-row_pos = 1.
ty_fieldcat-col_pos = 3.
ty_fieldcat-fieldname = 'AS4USER'.
ty_fieldcat-tabname = 'GT_DD02L'.
ty_fieldcat-coltext = 'CREATED'.
ty_fieldcat-outputlen = 10.
APPEND ty_fieldcat TO it_fieldcat.
CLEAR ty_fieldcat.
```

```
ty_fieldcat-row_pos = 1.
ty_fieldcat-col_pos = 4.
ty_fieldcat-fieldname = 'AS4DATE'.
ty_fieldcat-tabname = 'GT_DD02L'.
ty_fieldcat-coltext = 'DATE'.
ty_fieldcat-outputlen = 10.
APPEND ty_fieldcat TO it_fieldcat.
CLEAR ty_fieldcat.
```

```
ty_fieldcat-row_pos = 1.
ty_fieldcat-col_pos = 5.
ty_fieldcat-fieldname = 'AS4TIME'.
ty_fieldcat-tabname = 'GT_DD02L'.
ty_fieldcat-coltext = 'TIME'.
ty_fieldcat-outputlen = 10.
APPEND ty_fieldcat TO it_fieldcat.
CLEAR ty_fieldcat.
```

```
ty_fieldcat-row_pos = 1.
ty_fieldcat-col_pos = 6.
ty_fieldcat-fieldname = 'CONTFLAG'.
ty_fieldcat-tabname = 'GT_DD02L'.
ty_fieldcat-coltext = 'Delivery Class'.
ty_fieldcat-outputlen = 15.
APPEND ty_fieldcat TO it_fieldcat.
CLEAR ty_fieldcat.
```

```
ENDFORM.                " ALV_100_FIELDCAT
*&-----*
*&      Form  ALV_101_LAYOUT
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM alv_101_layout .
```

```
ty_lay2-grid_title = 'FIELDS'.
ty_lay2-zebra = 'X'.
```

```

ty_lay2-no_toolbar = 'X'.

ENDFORM.                    " ALV_101_LAYOUT
*&-----*
*&      Form  ALV_101_FIELDCAT
*&-----*
*      text
*-----*
* --> p1      text
* <-- p2      text
*-----*
FORM alv_101_fieldcat .

REFRESH it_fieldcat.
REFRESH it_fcat.
CLEAR ty_fcat.

ty_fcat-row_pos = 1.
ty_fcat-col_pos = 1.
ty_fcat-fieldname = 'FIELDNAME'.
ty_fcat-tabname = 'GT_DD03L'.
ty_fcat-coltext = 'Fieldname'.
ty_fcat-outputlen = 10.
APPEND ty_fcat TO it_fcat.

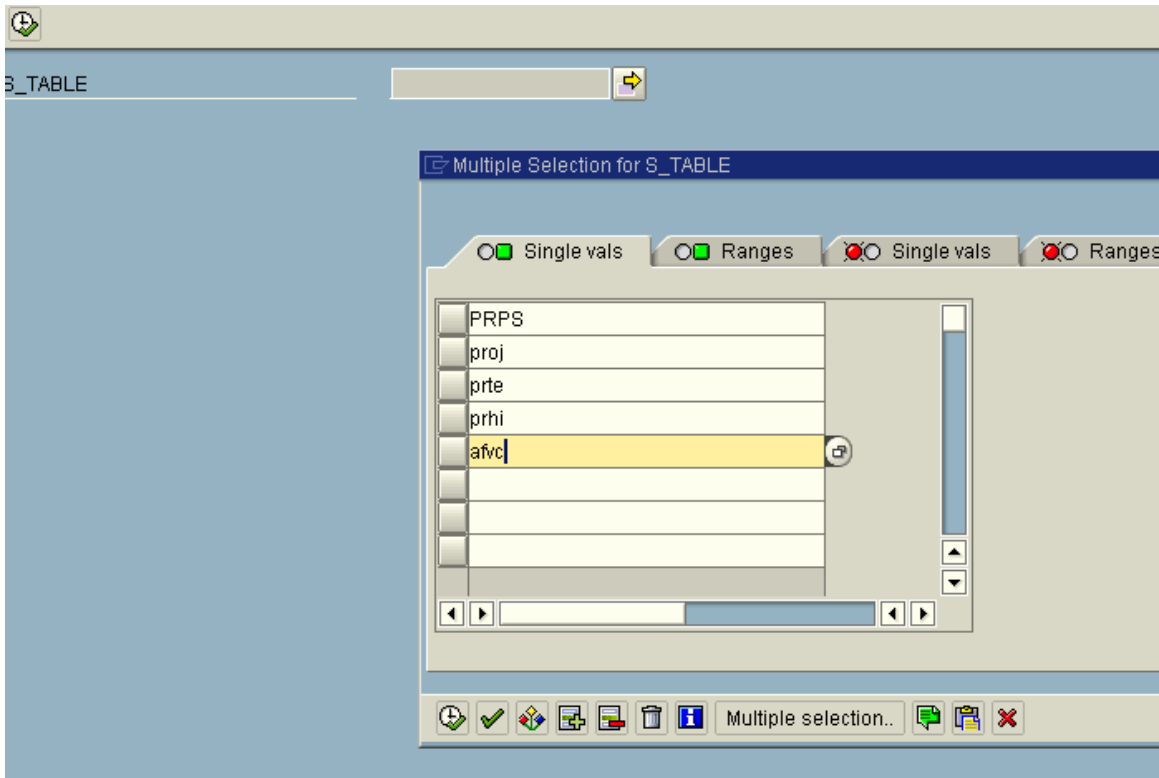
ty_fcat-row_pos = 1.
ty_fcat-col_pos = 2.
ty_fcat-fieldname = 'CHECKTABLE'.
ty_fcat-tabname = 'GT_DD03L'.
ty_fcat-coltext = 'CHECKTABLE'.
ty_fcat-outputlen = 10.
APPEND ty_fcat TO it_fcat.

ty_fcat-row_pos = 1.
ty_fcat-col_pos = 3.
ty_fcat-fieldname = 'KEYFLAG'.
ty_fcat-tabname = 'GT_DD03L'.
ty_fcat-coltext = 'Key Flag'.
ty_fcat-outputlen = 10.
APPEND ty_fcat TO it_fcat.

ENDFORM.                    " ALV_101_FIELDCAT

```

Selection screen



First Basic Output.

TABLES					
TableName	CATEGORY	CREATED	DATE	TIME	Delivery Class
AFVC	TRANSP	SAP	07.09.2005	01:14:34	A
PRHI	TRANSP	SAP	29.07.1998	18:55:34	A
PROJ	TRANSP	SAP	05.08.2003	16:42:31	A
PRPS	TRANSP	SAP	05.08.2003	16:43:48	A
PRTE	TRANSP	SAP	10.03.1999	17:41:25	A

On double clicking the table name we get the fields in the table as

FIELDS

Fieldname	CHECKTAB...	Key Flag
MANDT	T000	X
PSPNR		X
.INCLUDE		
PSPID		
.INCLUDE		
POST1		
OBJNR	ONR00	
ERNAM		
ERDAT		
AENAM		
AEDAT		
KMSK		
AUTOD		
STSPD	TJ20	
STSPR	TJ20	
VERNR	TCJ04	
VERNA		
ASTNR	TCJ05	
ASTNA		