

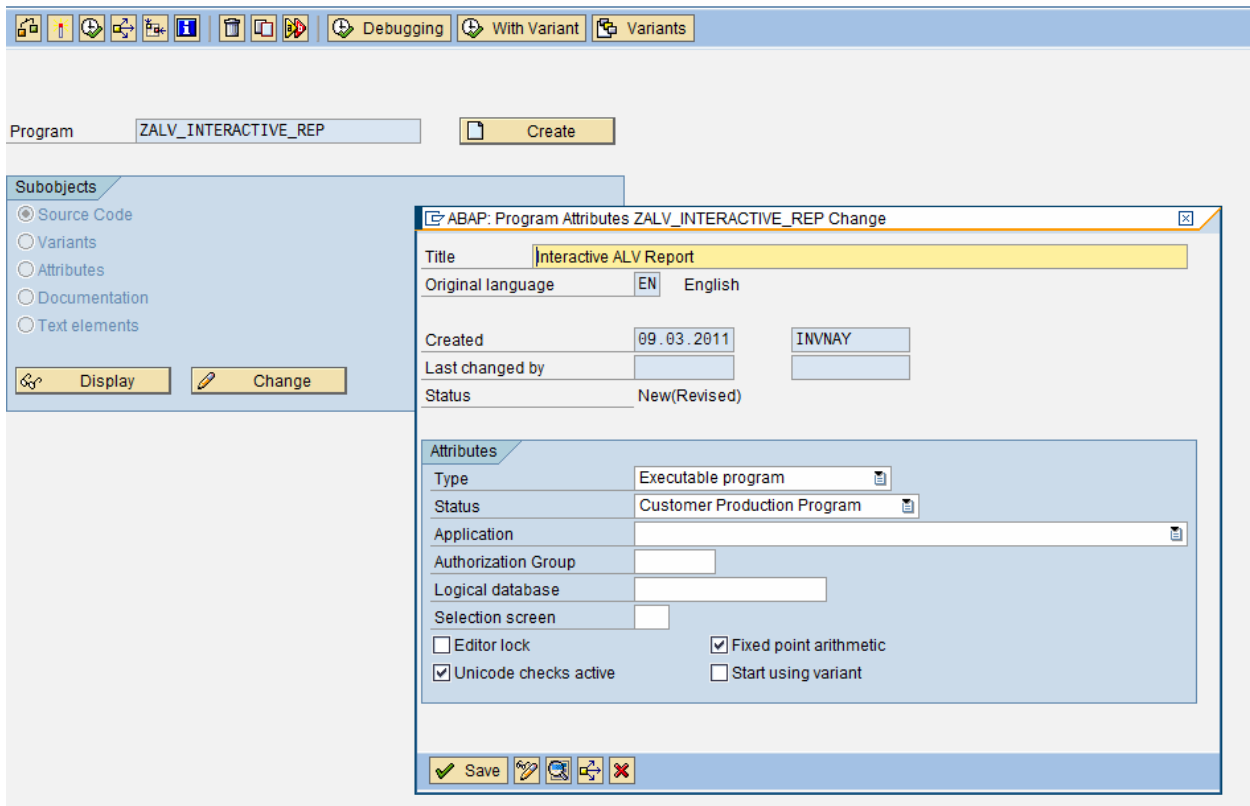
Scenario – This is a simple ALV list report in which the initial display consists of the header details of the Purchase order.

When the user double clicks on any of the PO in the list, it should show the item level details of that particular PO and other required data.

This is a simple interactive report.

- 1) Create the program with the following attributes

ABAP Editor: Initial Screen



- 2) The code for the program is as follows.



1 ALV Interactive
Report.txt

- 3) We initially have to do all the declarations required for the program.
- 4) Now for displaying the data in an interactive report we need to make sure, we have the data in an internal table that has to be displayed in the output. (perform data_retrieval)
- 5) And also the field catalogue which helps in displaying the output internal table (here it is it_ekko), which also holds all the required information about the columns. (perform build_fieldcatalog)

INITIALIZATION.

```
v_repid = sy-repid.  
PERFORM build_fieldcatlog.  
PERFORM event_call.  
PERFORM populate_event.
```

START-OF-SELECTION.

```
PERFORM data_retrieval.  
PERFORM build_listheader USING it_listheader.  
PERFORM display_alv_report.
```

..

- 6) Now we do the data retrieval code in perform data_retrieval.
- 7) We maintain all the characteristics of the columns in the field catalog
- 8) To use the events in the alv list, we use the standard function module to get all the available events in perform event_call and in 'perform populate_event', we will populate the name of the form that has to be executed when that particular event is called.
- 9) Here in this program, we are populating the event TOP_OF_PAGE with the form 'TOP_OF_PAGE' for the initial list and 'F_TOP_OF_PAGE' for the secondary list. And therefore we get different headings for both the list.
- 10) Now In the perform build_listheader, we give the header of the list in the forms mentioned in the above step 9.
- 11) For the event user_command, we gave the form 'USER_COMMAND' and inside this form, we give the interactive list program details.

```
*&-----*
*&      Form  USER_COMMAND
*&-----*
*      text
*-----*
*      -->R_UCOMM      text
*      -->,           text
*      -->RS_SLEFIELDtext
*-----*
FORM user_command USING r_ucomm LIKE sy-ucomm
rs_selfield TYPE slis_selfield.
CASE r_ucomm.
  WHEN '&IC1'.
    READ TABLE it_ekko INTO wa_ekko INDEX rs_selfield-tabindex.
    PERFORM build_fieldcatlog_ekpo.
    PERFORM event_call_ekpo.
    PERFORM populate_event_ekpo.
    PERFORM data_retrieval_ekpo.
    PERFORM build_listheader_ekpo USING it_listheader_1.
    PERFORM display_alv_ekpo.
ENDCASE.
ENDFORM.                    "user_command
*&-----*
```

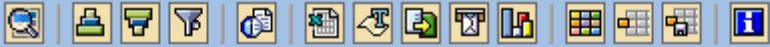
12) Here as seen above, we repeat the same process we did for displaying the first list.

13) While calling the Function module, REUSE_ALV_GRID_DISPLAY, we should make sure we populate the following parameters mandatorily.

- i_callback_program
- it_fieldcat
- t_outtab

14) Once executed, we get the following output.

Interactive ALV Report



This is a sample ALV Program

Basic List

PO NO.	DATE.	COMPANY ...	DOCUMENT ...	VENDOR C...
4500000000	03.07.200	MY01	NB	0007000383
4500000001	06.07.200	MY01	NB	0007000183
4500000002	06.07.200	MY01	NB	0007000926
4500000003	06.07.200	MY01	NB	0007000842
4500000004	07.07.200	MY01	NB	0007000832
4500000005	07.07.200	MY01	NB	0007000554
4500000006	07.07.200	MY01	NB	0007000707
4500000007	07.07.200	MY01	NB	0007000233
4500000008	07.07.200	MY01	NB	0007000589
4500000009	07.07.200	MY01	NB	0007001886
4500000010	07.07.200	MY01	NB	0007000619
4500000011	07.07.200	MY01	NB	0007000828
4500000012	07.07.200	MY01	NB	0007000947
4500000013	07.07.200	MY01	NB	0007001845
4500000014	07.07.200	MY01	NB	0007000912
4500000015	07.07.200	MY01	NB	0007000233
4500000016	07.07.200	MY01	NB	0007000690
4500000017	07.07.200	MY01	NB	0007000911
4500000018	07.07.200	MY01	NB	0007000563
4500000019	07.07.200	MY01	NB	0007000565
4500000020	07.07.200	MY01	NB	0007000770
4500000021	07.07.200	MY01	NB	0007001015
4500000022	07.07.200	MY01	NB	0007000440

15) When the user double clicks on any of the PO, it shows the item level details of the PO, as shown below.

Interactive ALV Report



Different Top Of Page Heading

First Secondary List

PO NO.	LINE NO	MATERIAL ...	QUANTITY	UOM	PRICE
4500000012	10	0000000000	2.000	EA	388.95
4500000012	20	0000000000	3.000	EA	2,101.60
4500000012	30	0000000000	3.000	EA	2,317.30
4500000012	40	0000000000	3.000	EA	2,317.30
4500000012	50	0000000000	2.000	EA	1,490.70