# ABAP Code

REPORT zmm\_po\_close.  
  
TABLES: ekko, ekpo, mara, lfa1.  
  
TYPES: BEGIN OF ty\_po,  
 ebeln TYPE ekko-ebeln,  
 matnr TYPE mara-matnr,  
 werks TYPE ekpo-werks,  
 lifnr TYPE lfa1-lifnr,  
 elikz TYPE ekpo-elikz,  
 END OF ty\_po.  
  
DATA: it\_po TYPE TABLE OF ty\_po,  
 wa\_po TYPE ty\_po.  
  
DATA: it\_selected TYPE STANDARD TABLE OF ty\_po,  
 wa\_selected TYPE ty\_po.  
  
DATA: o\_alv TYPE REF TO cl\_gui\_alv\_grid,  
 o\_container TYPE REF TO cl\_gui\_custom\_container.  
  
SELECT-OPTIONS: s\_matnr FOR wa\_po-matnr,  
 s\_werks FOR wa\_po-werks OBLIGATORY,  
 s\_lifnr FOR wa\_po-lifnr,  
 s\_mtart FOR mara-mtart.  
  
START-OF-SELECTION.  
  
 " Fetch materials based on material type  
 IF s\_mtart[] IS NOT INITIAL.  
 SELECT matnr FROM mara  
 INTO TABLE @DATA(it\_materials)  
 WHERE mtart IN s\_mtart.  
 ENDIF.  
  
 " Fetch purchase orders based on vendor  
 IF s\_lifnr[] IS NOT INITIAL.  
 SELECT ebeln FROM ekko  
 INTO TABLE @DATA(it\_po\_numbers)  
 WHERE lifnr IN s\_lifnr.  
 ENDIF.  
  
 " Fetch purchase order items based on selection criteria  
 SELECT ebeln, matnr, werks, lifnr, elikz  
 FROM ekpo  
 INTO TABLE it\_po  
 WHERE werks IN s\_werks  
 AND ( matnr IN s\_matnr OR  
 ( s\_mtart[] IS INITIAL AND s\_lifnr[] IS INITIAL ) )  
 AND elikz = ''.  
  
 " Display ALV output  
 PERFORM display\_alv.  
  
FORM display\_alv.  
 DATA: lt\_fieldcat TYPE lvc\_t\_fcat,  
 ls\_fieldcat TYPE lvc\_s\_fcat.  
  
 " Define field catalog for ALV  
 CLEAR ls\_fieldcat.  
 ls\_fieldcat-fieldname = 'EBELN'.  
 ls\_fieldcat-seltext\_m = 'PO Number'.  
 APPEND ls\_fieldcat TO lt\_fieldcat.  
  
 CLEAR ls\_fieldcat.  
 ls\_fieldcat-fieldname = 'MATNR'.  
 ls\_fieldcat-seltext\_m = 'Material Number'.  
 APPEND ls\_fieldcat TO lt\_fieldcat.  
  
 CLEAR ls\_fieldcat.  
 ls\_fieldcat-fieldname = 'WERKS'.  
 ls\_fieldcat-seltext\_m = 'Plant'.  
 APPEND ls\_fieldcat TO lt\_fieldcat.  
  
 CLEAR ls\_fieldcat.  
 ls\_fieldcat-fieldname = 'LIFNR'.  
 ls\_fieldcat-seltext\_m = 'Vendor'.  
 APPEND ls\_fieldcat TO lt\_fieldcat.  
  
 CLEAR ls\_fieldcat.  
 ls\_fieldcat-fieldname = 'ELIKZ'.  
 ls\_fieldcat-seltext\_m = 'Closed'.  
 APPEND ls\_fieldcat TO lt\_fieldcat.  
  
 " Create ALV container  
 CREATE OBJECT o\_container  
 EXPORTING  
 container\_name = 'ALV\_CONTAINER'.  
  
 " Create ALV Grid  
 CREATE OBJECT o\_alv  
 EXPORTING  
 i\_parent = o\_container.  
  
 " Set ALV data  
 CALL METHOD o\_alv->set\_table\_for\_first\_display  
 EXPORTING  
 i\_structure\_name = 'TY\_PO'  
 it\_fieldcatalog = lt\_fieldcat  
 CHANGING  
 it\_outtab = it\_po.  
  
 " Add selection option for closing POs  
 CALL METHOD o\_alv->register\_edit\_event  
 EXPORTING  
 i\_event\_id = cl\_gui\_alv\_grid=>mc\_evt\_modified.  
  
 " Handle user action for closing selected POs  
 DATA: lt\_selected TYPE STANDARD TABLE OF ty\_po.  
 CALL METHOD o\_alv->get\_selected\_rows  
 IMPORTING  
 et\_index\_rows = lt\_selected.  
  
 IF lt\_selected IS NOT INITIAL.  
 LOOP AT lt\_selected INTO wa\_selected.  
 READ TABLE it\_po INTO wa\_po INDEX wa\_selected-index.  
 IF sy-subrc = 0.  
 wa\_po-elikz = 'X'.  
 MODIFY it\_po INDEX wa\_selected-index TRANSPORTING elikz.  
 MODIFY ekpo FROM wa\_po.  
 ENDIF.  
 ENDLOOP.  
 COMMIT WORK.  
 MESSAGE 'Selected POs closed successfully.' TYPE 'S'.  
 ENDIF.  
  
ENDFORM.