ZSERVICE\_CONSUME

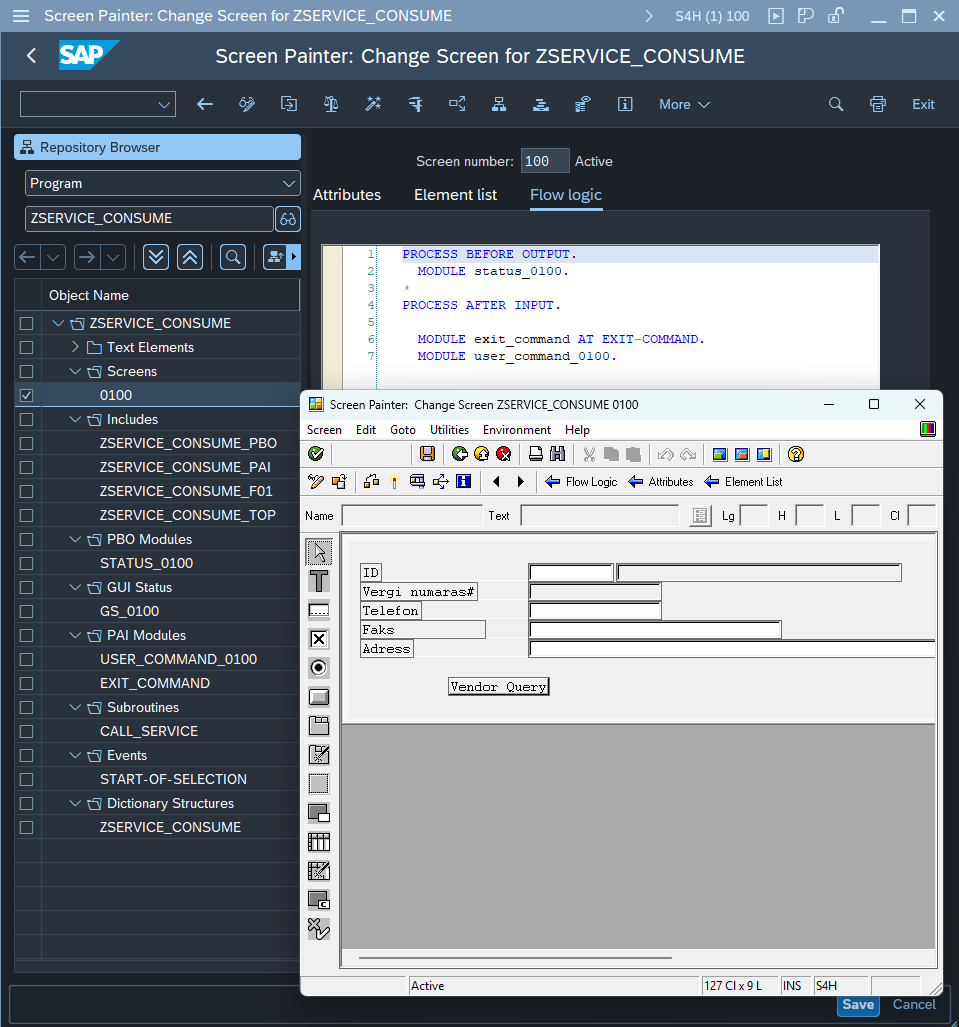
|  |
| --- |
| *\*&---------------------------------------------------------------------\** *\*& Report ZSERVICE\_CONSUME* *\*&---------------------------------------------------------------------\** *\*&* *\*&---------------------------------------------------------------------\** REPORT zservice\_consume.  INCLUDE zservice\_consume\_top. INCLUDE zservice\_consume\_pbo. INCLUDE zservice\_consume\_pai. INCLUDE zservice\_consume\_f01.  START-OF-SELECTION.   CALL SCREEN 0100. |

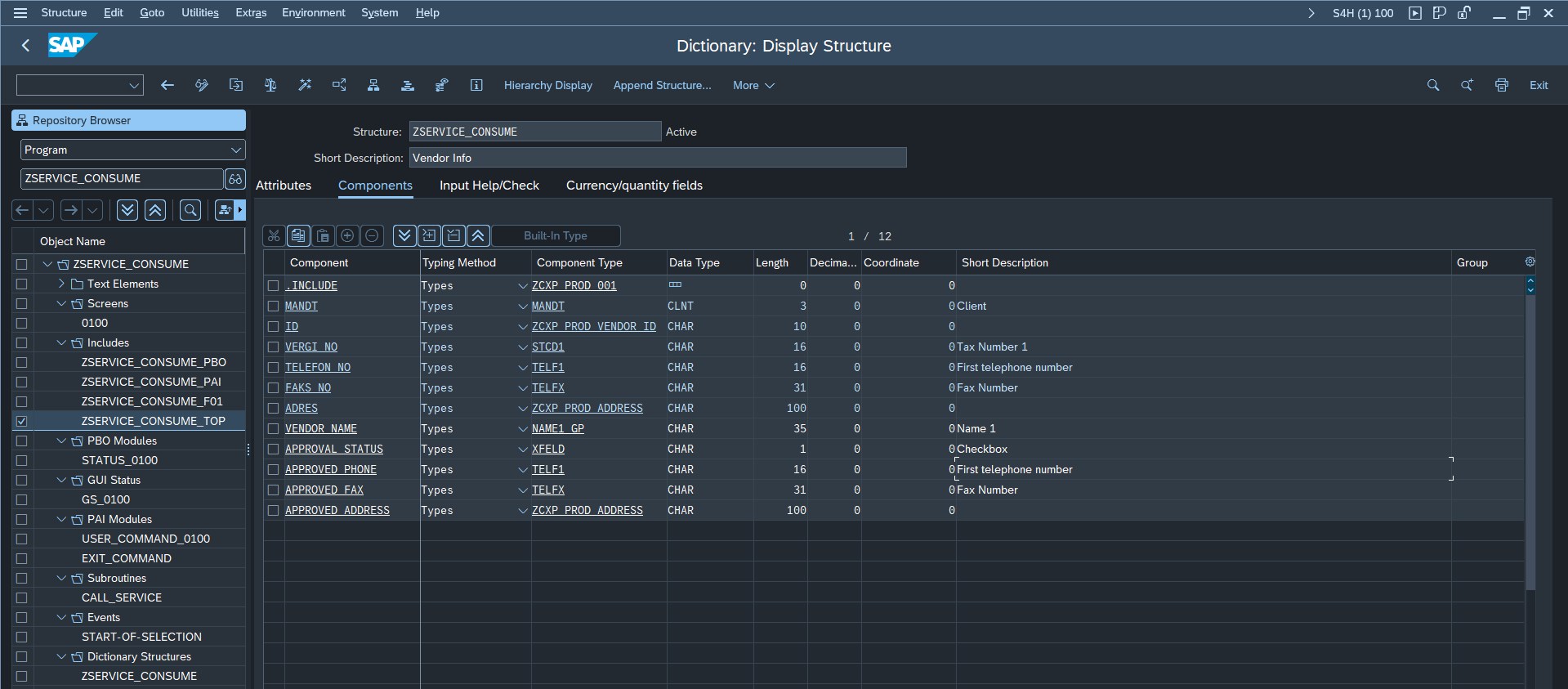
|  |
| --- |
| *\*&---------------------------------------------------------------------\** *\*& Include          ZSERVICE\_CONSUME\_TOP* *\*&---------------------------------------------------------------------\**  TABLES : zservice\_consume. |

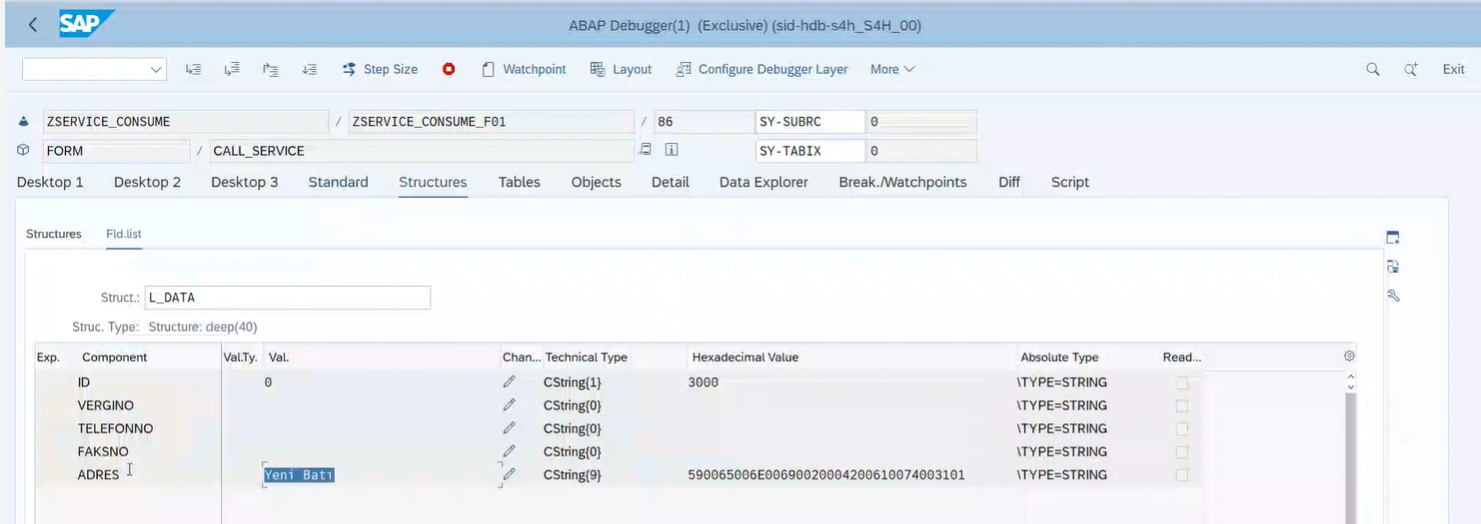
|  |
| --- |
| *\*----------------------------------------------------------------------\** *\*\*\*INCLUDE ZSERVICE\_CONSUME\_PBO.* *\*----------------------------------------------------------------------\** *\*&---------------------------------------------------------------------\** *\*& Module STATUS\_0100 OUTPUT* *\*&---------------------------------------------------------------------\** *\*&* *\*&---------------------------------------------------------------------\** MODULE status\_0100 OUTPUT.  SET PF-STATUS 'GS\_0100'. *\* SET TITLEBAR 'xxx'.*  *\* zservice\_consume-id = '1234'.*  ENDMODULE. |

|  |
| --- |
| *\*----------------------------------------------------------------------\** *\*\*\*INCLUDE ZSERVICE\_CONSUME\_PAI.* *\*----------------------------------------------------------------------\** *\*&---------------------------------------------------------------------\** *\*&      Module  USER\_COMMAND\_0100  INPUT* *\*&---------------------------------------------------------------------\** *\*       text* *\*----------------------------------------------------------------------\** MODULE user\_command\_0100 INPUT.    CASE sy-ucomm.     WHEN 'QUERY'.       PERFORM call\_service.  *\*      MESSAGE 'Query butonuna bastınız' TYPE 'I'.* *\*      LEAVE TO SCREEN 0.* *\*  WHEN 'EXIT'.* *\*    LEAVE TO SCREEN 0.* *\*  WHEN 'CANC'.* *\*    LEAVE TO SCREEN 0.* *\*    WHEN OTHERS.*   ENDCASE.  ENDMODULE. *\*&---------------------------------------------------------------------\** *\*&      Module  EXIT\_COMMAND  INPUT* *\*&---------------------------------------------------------------------\** *\*       text* *\*----------------------------------------------------------------------\** MODULE exit\_command INPUT.   LEAVE TO SCREEN 0. ENDMODULE. |

|  |
| --- |
| *\*&---------------------------------------------------------------------\** *\*& Include          ZSERVICE\_CONSUME\_F01* *\*&---------------------------------------------------------------------\**  FORM call\_service.  *\*  TYPES : BEGIN OF ty\_data,* *\*            id        TYPE string,* *\*            vergino   TYPE string,* *\*            telefonno TYPE string,* *\*            faksno    TYPE string,* *\*            adres     TYPE string,* *\*          END   OF ty\_data.*    DATA : lo\_http\_client TYPE REF TO if\_http\_client.    DATA : l\_data TYPE zservice\_consume. *\*       t\_data TYPE STANDARD TABLE OF ty\_data.*  *\*Data variables for storing response in xstring and string*   DATA  : lv\_string    TYPE string.  *\*Pass the URL to get Data*   lv\_string = 'http://10.1.4.21:50000/zcxp\_vendor\_srv/VENDOR'.    lv\_string = lv\_string && '?ID=' && zservice\_consume-id.  *\*Creation of New IF\_lo\_http\_client Object*   cl\_http\_client=>create\_by\_url(     EXPORTING       url                = lv\_string     IMPORTING       client             = lo\_http\_client     EXCEPTIONS       argument\_not\_found = 1       plugin\_not\_active  = 2       internal\_error     = 3   ).    IF sy-subrc IS NOT INITIAL. *\* Handle errors*   ENDIF.    CALL METHOD lo\_http\_client->send     EXCEPTIONS       http\_communication\_failure = 1       http\_invalid\_state         = 2       http\_processing\_failed     = 3       OTHERS                     = 4.    CALL METHOD lo\_http\_client->receive     EXCEPTIONS       http\_communication\_failure = 1       http\_invalid\_state         = 2       http\_processing\_failed     = 3       OTHERS                     = 4.    IF sy-subrc NE 0.      DATA subrc TYPE sysubrc.     DATA errortext TYPE string.      CALL METHOD lo\_http\_client->get\_last\_error       IMPORTING         code    = subrc         message = errortext.  *\*  WRITE: / 'communication\_error( receive )',* *\*         / 'code: ', subrc, 'message: ', errortext.*     EXIT.    ENDIF.   *\*Return the HTTP body of this entity as string data*   DATA(lv\_string2) = lo\_http\_client->response->get\_cdata( ).    /ui2/cl\_json=>deserialize(     EXPORTING       json        = lv\_string2       pretty\_name = /ui2/cl\_json=>pretty\_mode-camel\_case     CHANGING       data        = l\_data *\*     data        = t\_data*   ).  *\*  zservice\_consume-adres = l\_data-adres.*    MOVE-CORRESPONDING l\_data TO zservice\_consume.    SELECT SINGLE name1 AS vendor\_name FROM lfa1 INTO CORRESPONDING FIELDS OF zservice\_consume WHERE lifnr EQ zservice\_consume-id.    BREAK-POINT.  ENDFORM. |

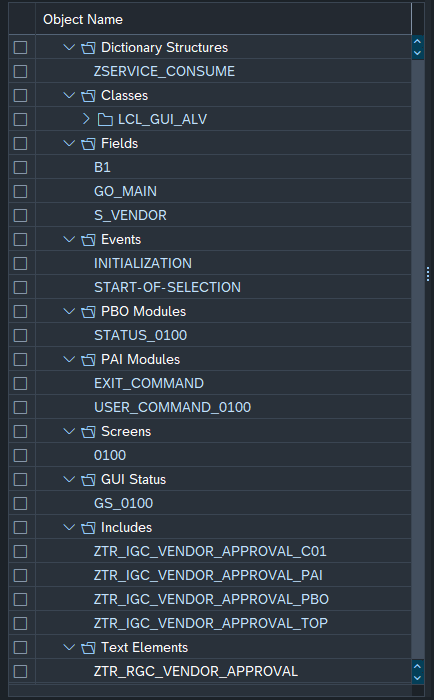






ZTR\_RGC\_VENDOR\_APPROVAL Programı (ZTR\_IGC\_VENDOR\_APPROVAL\_C01)

|  |
| --- |
| \*&---------------------------------------------------------------------\* \*& Report ZSERVICE\_CONSUME \*&---------------------------------------------------------------------\* \*& \*&---------------------------------------------------------------------\* REPORT ztr\_rgc\_vendor\_approval.  INCLUDE ZTR\_IGC\_VENDOR\_APPROVAL\_TOP. INCLUDE ZTR\_IGC\_VENDOR\_APPROVAL\_PBO. INCLUDE ZTR\_IGC\_VENDOR\_APPROVAL\_PAI. INCLUDE ZTR\_IGC\_VENDOR\_APPROVAL\_C01. INITIALIZATION.    CREATE OBJECT go\_main.  START-OF-SELECTION.    CALL SCREEN 0100. |

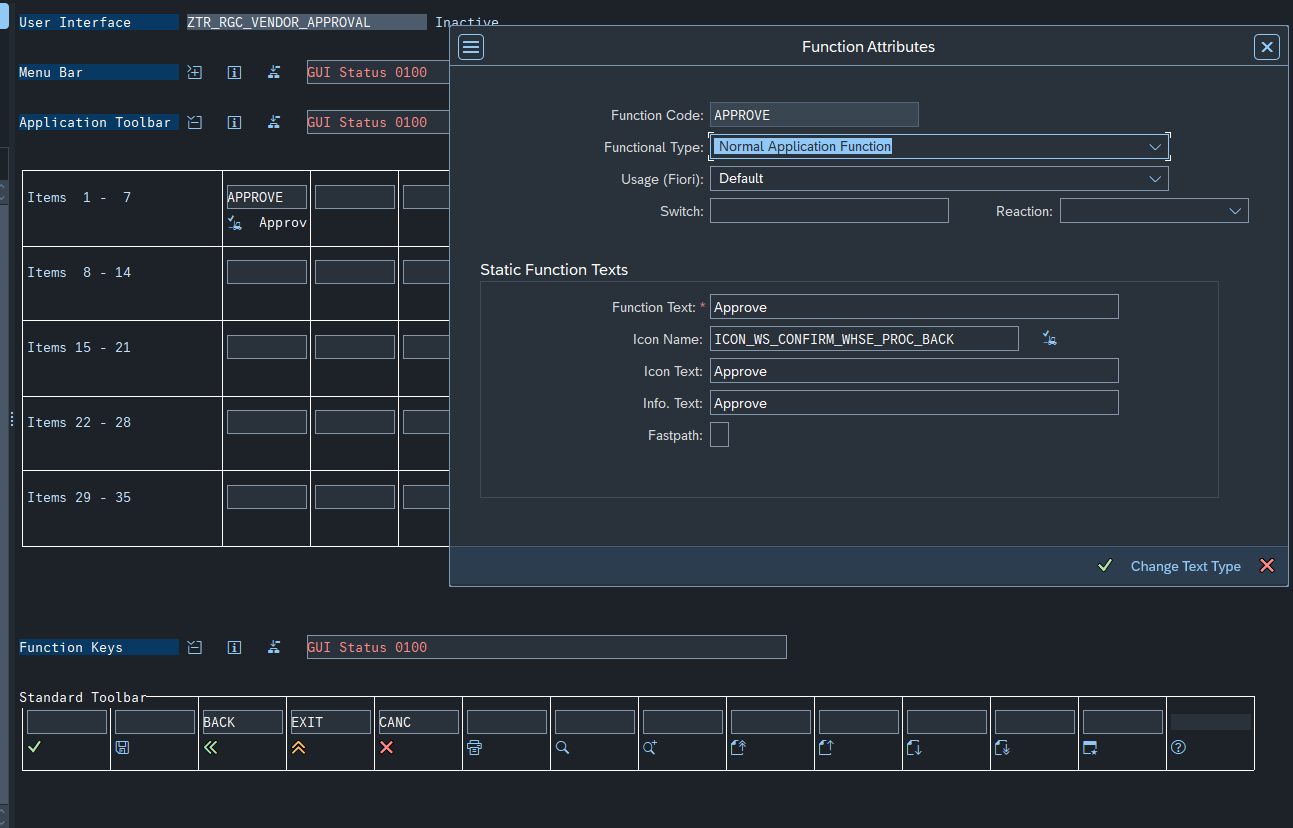


|  |
| --- |
| \*&---------------------------------------------------------------------\* \*& Include          ZTR\_I\_VENDOR\_QUERY\_C01 \*&---------------------------------------------------------------------\* CLASS lcl\_gui\_alv IMPLEMENTATION.    METHOD user\_command .      CASE sy-ucomm.       WHEN gc\_user\_command\_approve.         CALL METHOD approve.     ENDCASE.   ENDMETHOD.    METHOD init\_0100.      SET PF-STATUS 'GS\_0100'.      build\_data( ).      init\_grid( ).      display( ).    ENDMETHOD.    METHOD approve .      mo\_grid->get\_selected\_rows(       IMPORTING         et\_index\_rows = DATA(lt\_index)         et\_row\_no     = DATA(lt\_row)     ).      BREAK-POINT.      LOOP AT lt\_index INTO DATA(ls\_index).        READ TABLE mt\_out ASSIGNING FIELD-SYMBOL(<lfs\_out>) INDEX ls\_index-index.        IF sy-subrc EQ 0.          <lfs\_out>-approval\_status = abap\_true.        ENDIF.      ENDLOOP.      mo\_grid->refresh\_table\_display( \*      EXPORTING \*        is\_stable      = \*        i\_soft\_refresh =       EXCEPTIONS         finished = 1         OTHERS   = 2     ).    ENDMETHOD.    METHOD build\_data .      SELECT \* FROM zcxp\_prod\_002 INTO CORRESPONDING FIELDS OF TABLE mt\_out WHERE id IN s\_vendor.      LOOP AT mt\_out ASSIGNING FIELD-SYMBOL(<lfs\_out>).        SELECT SINGLE name1 FROM lfa1 INTO <lfs\_out>-vendor\_name WHERE lifnr EQ <lfs\_out>-id.      ENDLOOP.      APPEND LINES OF mt\_out TO mt\_out.     APPEND LINES OF mt\_out TO mt\_out.    ENDMETHOD.    METHOD set\_title.      READ TABLE mt\_field\_catalog ASSIGNING FIELD-SYMBOL(<lfs\_field\_catalog>) WITH KEY fieldname = im\_field\_name.      IF sy-subrc EQ 0.       <lfs\_field\_catalog>-reptext =       <lfs\_field\_catalog>-scrtext\_s =       <lfs\_field\_catalog>-scrtext\_m =       <lfs\_field\_catalog>-scrtext\_l = im\_title.     ENDIF.    ENDMETHOD.    METHOD build\_field\_catalog.   \* Filling the fieldcatalog table     CALL FUNCTION 'LVC\_FIELDCATALOG\_MERGE'       EXPORTING         i\_structure\_name       = gc\_structure\_name       CHANGING         ct\_fieldcat            = mt\_field\_catalog       EXCEPTIONS         inconsistent\_interface = 1         program\_error          = 2         OTHERS                 = 3.      LOOP AT mt\_field\_catalog ASSIGNING FIELD-SYMBOL(<lfs\_fcat>).        <lfs\_fcat>-colddictxt = 'L'.        IF <lfs\_fcat>-fieldname EQ 'APPROVAL\_STATUS'.         <lfs\_fcat>-checkbox = abap\_true.       ENDIF.      ENDLOOP.      set\_title(       im\_field\_name = 'ID'       im\_title      = TEXT-h01     ).      set\_title(       im\_field\_name = 'ADRES'       im\_title      = TEXT-h02     ).      set\_title(       im\_field\_name = 'VENDOR\_NAME'       im\_title      = TEXT-h03     ).      set\_title(       im\_field\_name = 'APPROVAL\_STATUS'       im\_title      = TEXT-h04     ).    ENDMETHOD.    METHOD get\_container.      DATA : lo\_docking\_container TYPE REF TO cl\_gui\_docking\_container.      IF lo\_docking\_container IS INITIAL.        CREATE OBJECT lo\_docking\_container         EXPORTING           side                        = cl\_gui\_docking\_container=>dock\_at\_top           extension                   = 1000 \*         ratio                       = 95           caption                     = 'ALV List '         EXCEPTIONS           cntl\_error                  = 1           cntl\_system\_error           = 2           create\_error                = 3           lifetime\_error              = 4           lifetime\_dynpro\_dynpro\_link = 5           OTHERS                      = 6.     ENDIF.      re\_container\_object = lo\_docking\_container.    ENDMETHOD.    METHOD display.      DATA : ls\_variant TYPE disvariant,            ls\_layout  TYPE lvc\_s\_layo.      ls\_variant-report = sy-repid.      ls\_layout-zebra      = abap\_on.     ls\_layout-cwidth\_opt = abap\_on.     ls\_layout-sel\_mode   = 'A'.      CALL METHOD mo\_grid->set\_table\_for\_first\_display       EXPORTING         is\_variant      = ls\_variant         i\_save          = 'A'         is\_layout       = ls\_layout       CHANGING         it\_outtab       = mt\_out         it\_fieldcatalog = mt\_field\_catalog.    ENDMETHOD.    METHOD init\_grid.      CALL METHOD build\_field\_catalog.      CALL METHOD get\_container       RECEIVING         re\_container\_object = DATA(lo\_container).      " Create grid object     CREATE OBJECT mo\_grid       EXPORTING         i\_parent = lo\_container.    ENDMETHOD.  ENDCLASS. |

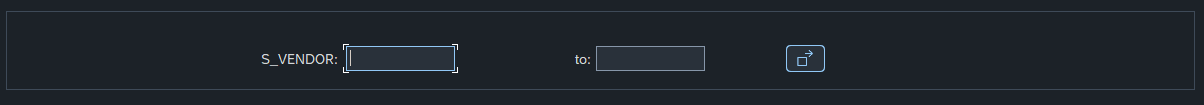
|  |
| --- |
| \*----------------------------------------------------------------------\* \*\*\*INCLUDE ZTR\_I\_VENDOR\_QUERY\_PAI \*----------------------------------------------------------------------\* \*&---------------------------------------------------------------------\* \*&      Module  USER\_COMMAND\_0100  INPUT \*&---------------------------------------------------------------------\* \*       text \*----------------------------------------------------------------------\* MODULE user\_command\_0100 INPUT.    CALL METHOD go\_main->user\_command.  ENDMODULE. \*&---------------------------------------------------------------------\* \*&      Module  EXIT\_COMMAND  INPUT \*&---------------------------------------------------------------------\* \*       text \*----------------------------------------------------------------------\* MODULE exit\_command INPUT.   LEAVE TO SCREEN 0. ENDMODULE. |

|  |
| --- |
| \*----------------------------------------------------------------------\* \*\*\*INCLUDE ZTR\_I\_VENDOR\_QUERY\_PBO. \*----------------------------------------------------------------------\* \*&---------------------------------------------------------------------\* \*& Module STATUS\_0100 OUTPUT \*&---------------------------------------------------------------------\* \*& \*&---------------------------------------------------------------------\* MODULE status\_0100 OUTPUT.   go\_main->init\_0100( ). ENDMODULE. |

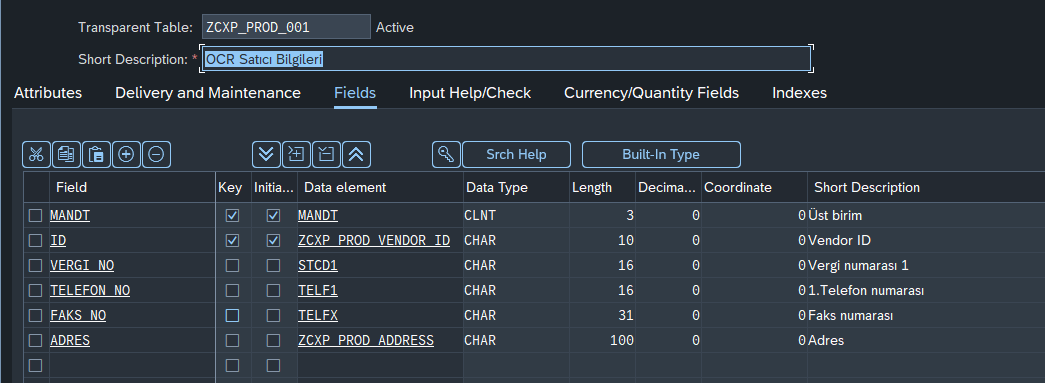
|  |
| --- |
| \*&---------------------------------------------------------------------\* \*& Include          ZTR\_I\_VENDOR\_APPROVAL\_TOP \*&---------------------------------------------------------------------\*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \* GLOBAL DATA DEFINITIONS                                              \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*----------------------------------------------------------------------\* \* INCLUDE - definitions                                                \* \*----------------------------------------------------------------------\* \*----------------------------------------------------------------------\* \* CONSTANT - definitions                                               \* \*----------------------------------------------------------------------\*  \*----------------------------------------------------------------------\* \* TYPE - definitions                                                   \* \*----------------------------------------------------------------------\* \*----------------------------------------------------------------------\* \* DDIC - TABLE / STRUCTURE / VIEW definitions                          \* \*----------------------------------------------------------------------\* TABLES : zservice\_consume. \*----------------------------------------------------------------------\* \* RANGE definitions                                                    \* \*----------------------------------------------------------------------\* \*----------------------------------------------------------------------\* \* REFERENCE definitions                                                \* \*----------------------------------------------------------------------\*  CLASS lcl\_gui\_alv DEFINITION.    PUBLIC SECTION.      CONSTANTS : gc\_user\_command\_approve TYPE sy-ucomm VALUE 'APPROVE',                 gc\_structure\_name       TYPE typename VALUE 'ZSERVICE\_CONSUME'.      DATA : mo\_grid TYPE REF TO cl\_gui\_alv\_grid.      DATA : mt\_out           TYPE TABLE OF zservice\_consume,            mt\_field\_catalog TYPE lvc\_t\_fcat.      METHODS :       build\_data,       build\_field\_catalog,       init\_0100,       user\_command,       get\_container RETURNING VALUE(re\_container\_object) TYPE REF TO cl\_gui\_docking\_container,       init\_grid,       display,       set\_title IMPORTING im\_field\_name TYPE lvc\_fname im\_title TYPE reptext,       approve.  ENDCLASS.  DATA : go\_main TYPE REF TO lcl\_gui\_alv.  \*----------------------------------------------------------------------\* \* INTERNAL TABLE / STRUCTURE definitions \*----------------------------------------------------------------------\* \*----------------------------------------------------------------------\* \* OTHER GLOBAL DATA definitions                                        \* \*----------------------------------------------------------------------\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \* GLOBAL DATA DEFINITIONS - END                                        \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \* SELECTION SCREENS                                                    \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  SELECTION-SCREEN BEGIN OF BLOCK b1 WITH FRAME.    SELECT-OPTIONS : s\_vendor FOR zservice\_consume-id.  SELECTION-SCREEN END OF BLOCK b1.  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \* SELECTION SCREENS - END                                              \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* |



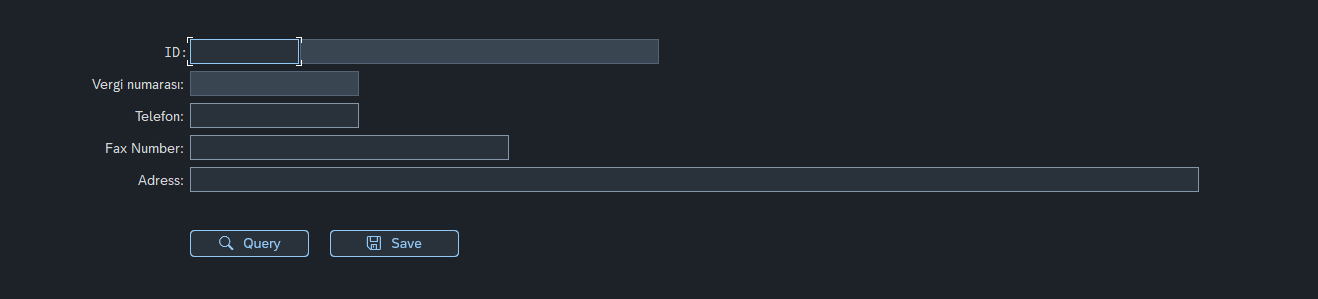
|  |
| --- |
| PROCESS BEFORE OUTPUT.    MODULE status\_0100. \* PROCESS AFTER INPUT.    MODULE exit\_command AT EXIT-COMMAND.    MODULE user\_command\_0100. |



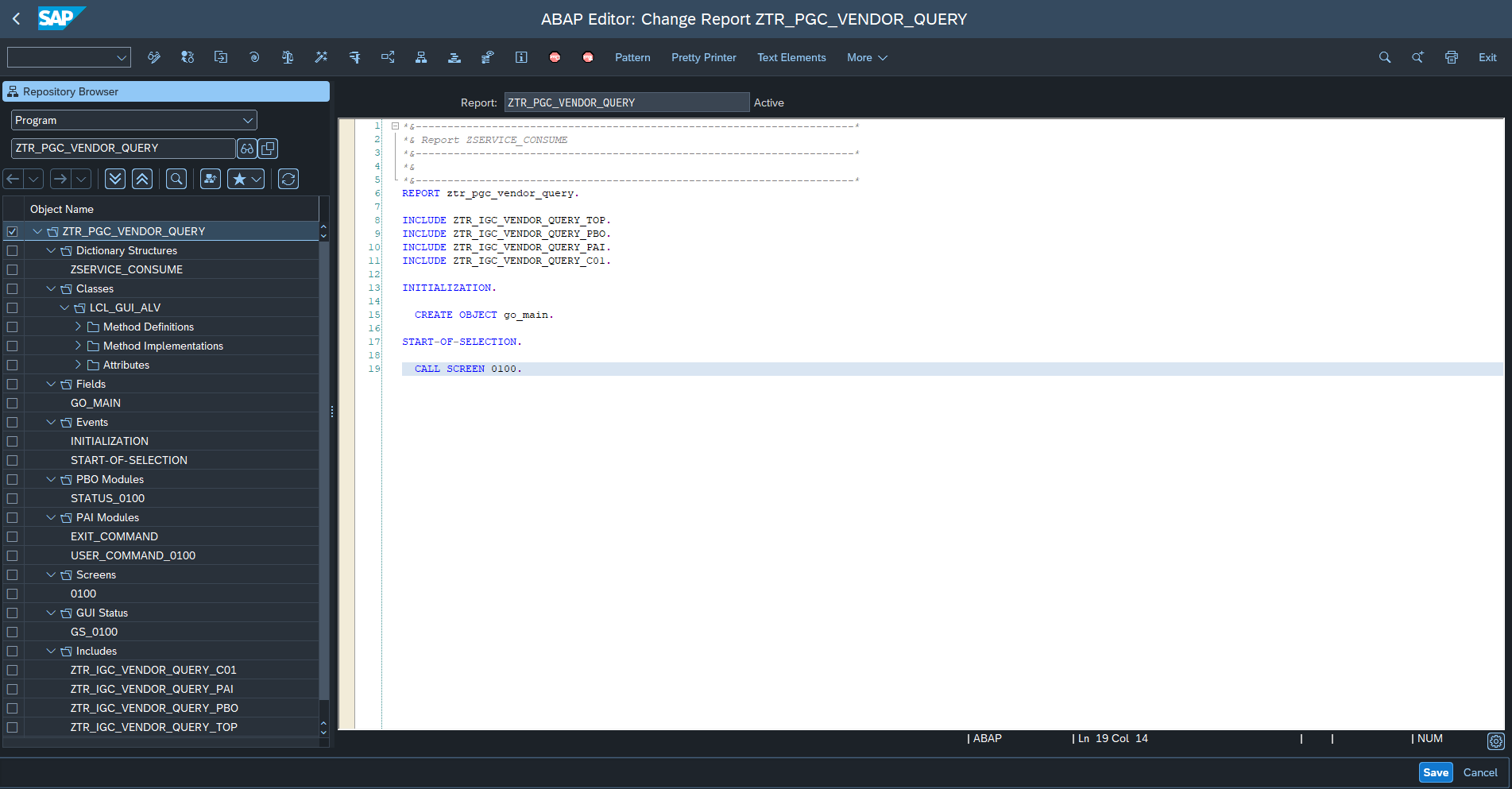
-

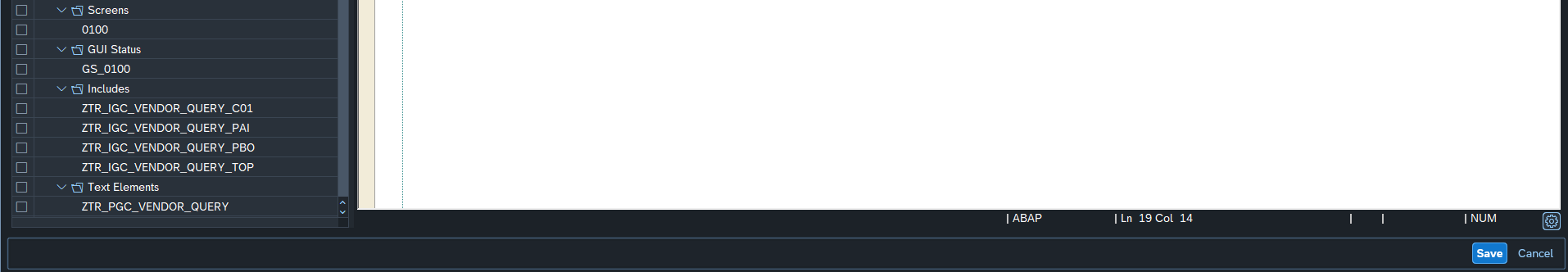


ZTR\_P\_VENDOR\_QUERY Programı



|  |
| --- |
| *\*&---------------------------------------------------------------------\** *\*& Report ZSERVICE\_CONSUME* *\*&---------------------------------------------------------------------\** *\*&* *\*&---------------------------------------------------------------------\** REPORT ztr\_pgc\_vendor\_query.  INCLUDE ZTR\_IGC\_VENDOR\_QUERY\_TOP. INCLUDE ZTR\_IGC\_VENDOR\_QUERY\_PBO. INCLUDE ZTR\_IGC\_VENDOR\_QUERY\_PAI. INCLUDE ZTR\_IGC\_VENDOR\_QUERY\_C01.  INITIALIZATION.    CREATE OBJECT go\_main.  START-OF-SELECTION.    CALL SCREEN 0100. |





|  |
| --- |
| *\*&---------------------------------------------------------------------\** *\*& Include          ZTR\_I\_VENDOR\_QUERY\_TOP* *\*&---------------------------------------------------------------------\**  *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** *\* GLOBAL DATA DEFINITIONS                                              \** *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** *\*----------------------------------------------------------------------\** *\* INCLUDE - definitions                                                \** *\*----------------------------------------------------------------------\** *\*----------------------------------------------------------------------\** *\* CONSTANT - definitions                                               \** *\*----------------------------------------------------------------------\**  *\*CONSTANTS : gc\_user\_command\_query TYPE sy-ucomm VALUE 'QUERY'.* *\*CONSTANTS : gc\_structure\_name TYPE typename VALUE 'ZHR\_S\_ALV'.*  *\*----------------------------------------------------------------------\** *\* TYPE - definitions                                                   \** *\*----------------------------------------------------------------------\** *\*----------------------------------------------------------------------\** *\* DDIC - TABLE / STRUCTURE / VIEW definitions                          \** *\*----------------------------------------------------------------------\** TABLES : zservice\_consume. *\*----------------------------------------------------------------------\** *\* RANGE definitions                                                    \** *\*----------------------------------------------------------------------\** *\*----------------------------------------------------------------------\** *\* REFERENCE definitions                                                \** *\*----------------------------------------------------------------------\**  CLASS lcl\_gui\_alv DEFINITION.    PUBLIC SECTION.      CONSTANTS : gc\_user\_command\_query TYPE sy-ucomm VALUE 'QUERY',                 gc\_user\_command\_save  TYPE sy-ucomm VALUE 'SAVE',                 gc\_endpoint           TYPE string VALUE 'http://10.1.4.21:50000/zcxp\_vendor\_srv/VENDOR'.  *\*    DATA : mt\_out           TYPE TABLE OF zhr\_s\_alv.*      METHODS : *\*      build\_data,* *\*      build\_alv\_data IMPORTING im\_pernr\_tab TYPE hrahq\_pernr\_table,* *\*      build\_data IMPORTING im\_pernr\_tab TYPE hrahq\_pernr\_table,* *\*      set\_text\_data CHANGING ch\_item TYPE zhr\_s\_alv,*       init\_0100,       user\_command,       vendor\_query,       save. *\*      insert\_log IMPORTING im\_personnel\_no TYPE zhr\_01-personel\_no im\_request TYPE zhr... im\_response TYPE zhr... im\_exception\_text TYPE string.*  ENDCLASS.  DATA : go\_main TYPE REF TO lcl\_gui\_alv.  *\*----------------------------------------------------------------------\** *\* INTERNAL TABLE / STRUCTURE definitions* *\*----------------------------------------------------------------------\** *\*----------------------------------------------------------------------\** *\* OTHER GLOBAL DATA definitions                                        \** *\*----------------------------------------------------------------------\** *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** *\* GLOBAL DATA DEFINITIONS - END                                        \** *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** *\* SELECTION SCREENS                                                    \** *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\** |

|  |
| --- |
| *\*----------------------------------------------------------------------\** *\*\*\*INCLUDE ZTR\_I\_VENDOR\_QUERY\_PBO.* *\*----------------------------------------------------------------------\** *\*&---------------------------------------------------------------------\** *\*& Module STATUS\_0100 OUTPUT* *\*&---------------------------------------------------------------------\** *\*&* *\*&---------------------------------------------------------------------\** MODULE status\_0100 OUTPUT.   go\_main->init\_0100( ). ENDMODULE. |

|  |
| --- |
| *\*----------------------------------------------------------------------\** *\*\*\*INCLUDE ZTR\_I\_VENDOR\_QUERY\_PAI* *\*----------------------------------------------------------------------\** *\*&---------------------------------------------------------------------\** *\*&      Module  USER\_COMMAND\_0100  INPUT* *\*&---------------------------------------------------------------------\** *\*       text* *\*----------------------------------------------------------------------\** MODULE user\_command\_0100 INPUT.    CALL METHOD go\_main->user\_command.  ENDMODULE. *\*&---------------------------------------------------------------------\** *\*&      Module  EXIT\_COMMAND  INPUT* *\*&---------------------------------------------------------------------\** *\*       text* *\*----------------------------------------------------------------------\** MODULE exit\_command INPUT.   LEAVE TO SCREEN 0. ENDMODULE. |

|  |
| --- |
| *\*&---------------------------------------------------------------------\** *\*& Include          ZTR\_I\_VENDOR\_QUERY\_C01* *\*&---------------------------------------------------------------------\** CLASS lcl\_gui\_alv IMPLEMENTATION.    METHOD user\_command .      CASE sy-ucomm.       WHEN gc\_user\_command\_query.         CALL METHOD vendor\_query.       WHEN gc\_user\_command\_save.         CALL METHOD save.     ENDCASE.   ENDMETHOD.    METHOD init\_0100.      SET PF-STATUS 'GS\_0100'.    ENDMETHOD.    METHOD vendor\_query .      DATA : lo\_http\_client TYPE REF TO if\_http\_client.      DATA : ls\_response\_data TYPE zservice\_consume.      DATA : lv\_endpoint   TYPE string,            lv\_subrc      TYPE sysubrc,            lv\_error\_text TYPE string.  *\*Pass the URL to get Data*     lv\_endpoint = gc\_endpoint && '?ID=' && zservice\_consume-id.  *\*Creation of New IF\_lo\_http\_client Object*     cl\_http\_client=>create\_by\_url(       EXPORTING         url                = lv\_endpoint       IMPORTING         client             = lo\_http\_client       EXCEPTIONS         argument\_not\_found = 1         plugin\_not\_active  = 2         internal\_error     = 3     ).      CALL METHOD lo\_http\_client->send       EXCEPTIONS         http\_communication\_failure = 1         http\_invalid\_state         = 2         http\_processing\_failed     = 3         OTHERS                     = 4.      CALL METHOD lo\_http\_client->receive       EXCEPTIONS         http\_communication\_failure = 1         http\_invalid\_state         = 2         http\_processing\_failed     = 3         OTHERS                     = 4.      *"""""""""""""*     *" Verinin HTTP cevabında mevcut olup olmadığını kontrol et*     IF ls\_response\_data IS INITIAL.       MESSAGE 'Vendor ID ' && zservice\_consume-id && ' not found. Please check.' TYPE 'I'.       RETURN.     ENDIF.     *""""""""""""""""""*      IF sy-subrc NE 0.        CALL METHOD lo\_http\_client->get\_last\_error         IMPORTING           code    = lv\_subrc           message = lv\_error\_text.        DATA(lv\_error) = 'communication\_error( receive )' && ` ` &&                        'code:' && ` ` &&                        lv\_subrc && ` ` &&                        'message:' && ` ` &&                        lv\_error\_text.        MESSAGE lv\_error TYPE 'I'.        EXIT.      ENDIF.  *\*Return the HTTP body of this entity as string data*     DATA(lv\_json) = lo\_http\_client->response->get\_cdata( ).      /ui2/cl\_json=>deserialize(       EXPORTING         json        = lv\_json         pretty\_name = /ui2/cl\_json=>pretty\_mode-camel\_case       CHANGING         data        = ls\_response\_data     ).      MOVE-CORRESPONDING ls\_response\_data TO zservice\_consume.      SELECT SINGLE name1 *\*    SELECT SINGLE name1 AS vendor\_name*       FROM lfa1       INTO zservice\_consume-vendor\_name *\*      INTO CORRESPONDING FIELDS OF zservice\_consume*      WHERE lifnr EQ zservice\_consume-id.    ENDMETHOD.    METHOD save.      DATA : ls\_db TYPE zcxp\_prod\_002.      MOVE-CORRESPONDING zservice\_consume TO ls\_db.      MODIFY zcxp\_prod\_002 FROM ls\_db.      IF sy-subrc EQ 0.       MESSAGE TEXT-s01 TYPE 'I'.     ENDIF.    ENDMETHOD. ENDCLASS. |

