# Приложения

## Листинг . Функциональный модуль ZBC\_BIW\_GET\_SCR\_STX\_TRAN

FUNCTION zbc\_biw\_get\_scr\_stx\_tran.
\*"----------------------------------------------------------------------
\*"  IMPORTING
\*"     VALUE(I\_REQUNR) TYPE  SRSC\_S\_IF\_SIMPLE-REQUNR
\*"     VALUE(I\_DSOURCE) TYPE  SRSC\_S\_IF\_SIMPLE-DSOURCE OPTIONAL
\*"     VALUE(I\_MAXSIZE) TYPE  SRSC\_S\_IF\_SIMPLE-MAXSIZE OPTIONAL
\*"     VALUE(I\_INITFLAG) TYPE  SRSC\_S\_IF\_SIMPLE-INITFLAG OPTIONAL
\*"     VALUE(I\_READ\_ONLY) TYPE  SRSC\_S\_IF\_SIMPLE-READONLY OPTIONAL
\*"     VALUE(I\_REMOTE\_CALL) TYPE  SBIWA\_FLAG DEFAULT SBIWA\_C\_FLAG\_OFF
\*"  TABLES
\*"      I\_T\_SELECT TYPE  SRSC\_T\_SELECT OPTIONAL
\*"      I\_T\_FIELDS TYPE  SRSC\_T\_FIELDS OPTIONAL
\*"      E\_T\_DATA STRUCTURE  ZCA\_SCR\_TX\_LINE OPTIONAL
\*"  EXCEPTIONS
\*"      NO\_MORE\_DATA
\*"      ERROR\_PASSED\_TO\_MESS\_HANDLER
\*"----------------------------------------------------------------------

  TYPES:

      BEGIN OF ty\_std\_text\_line,
          tdobject TYPE stxl-tdobject,
          tdname TYPE stxl-tdname,
          tdid TYPE stxl-tdid,
          clustr TYPE stxl-clustr,
          clustd TYPE stxl-clustd,
      END OF ty\_std\_text\_line,

      BEGIN OF ty\_std\_text\_line\_cluster,
         clustr TYPE stxl-clustr,
         clustd TYPE stxl-clustd,
      END OF ty\_std\_text\_line\_cluster.

  FIELD-SYMBOLS:
      <select> TYPE srsc\_s\_select,
      <std\_text\_line> TYPE ty\_std\_text\_line,
      <std\_text\_line\_cluster> TYPE tline.

  CONSTANTS:
      strandart\_text TYPE relid VALUE 'TX'.

  STATICS:
      datapakage TYPE srsc\_s\_if\_simple,
      counter\_datapakage TYPE i,
      db\_cursor TYPE cursor.

  DATA:
      std\_text\_raw TYPE STANDARD TABLE OF ty\_std\_text\_line,
      std\_text\_lines TYPE STANDARD TABLE OF tline,
      lines\_cluster TYPE STANDARD TABLE OF ty\_std\_text\_line\_cluster.

  DATA:
      line\_cluster TYPE ty\_std\_text\_line\_cluster,
      data\_line TYPE zca\_scr\_tx\_line,
      counter\_lines TYPE num\_rec.

  DATA:
      text\_names TYPE RANGE OF ty\_std\_text\_line-tdname,
      text\_ids TYPE RANGE OF ty\_std\_text\_line-tdname,
      text\_objects TYPE RANGE OF ty\_std\_text\_line-tdname.

  DATA:
      text\_name LIKE LINE OF text\_names,
      text\_id LIKE LINE OF text\_names,
      text\_object LIKE LINE OF text\_names.

  IF i\_initflag = sbiwa\_c\_flag\_on.

\* Check DataSource validity
    CASE i\_dsource.
      WHEN 'ZCA\_SCR\_TX\_TRAN'.
      WHEN OTHERS.
        RAISE error\_passed\_to\_mess\_handler.
    ENDCASE.

    datapakage-requnr    = i\_requnr.
    datapakage-dsource = i\_dsource.
    datapakage-maxsize   = i\_maxsize.

    APPEND LINES OF i\_t\_fields TO datapakage-t\_fields.
    APPEND LINES OF i\_t\_select TO datapakage-t\_select.

  ELSE.                 "Initialization mode or data extraction ?

    IF counter\_datapakage = 0.

      LOOP AT datapakage-t\_select ASSIGNING <select>.
        CASE <select>-fieldnm.
          WHEN 'TEXT\_OBJ'.
            MOVE-CORRESPONDING <select> TO text\_object.
            APPEND text\_object TO text\_objects.
          WHEN 'TEXT\_ID'.
            MOVE-CORRESPONDING <select> TO text\_id.
            APPEND text\_id TO text\_ids.
          WHEN 'TEXT\_NAME'.
            MOVE-CORRESPONDING <select> TO text\_name.
            APPEND text\_name TO text\_names.
          WHEN OTHERS.
        ENDCASE.
      ENDLOOP.

\*   First data package -> OPEN CURSOR
      OPEN CURSOR WITH HOLD db\_cursor FOR
          SELECT
              tdobject
              tdname
              tdid
              clustr
              clustd
          FROM stxl
          WHERE
              relid    = strandart\_text
              AND tdobject IN text\_objects
              AND tdname   IN text\_names
              AND tdid     IN text\_ids
              AND tdspras  = sy-langu
              AND srtf2 = 0.

    ENDIF.   "First data package ?

    FETCH NEXT CURSOR db\_cursor
               INTO CORRESPONDING FIELDS
               OF TABLE std\_text\_raw
               PACKAGE SIZE datapakage-maxsize.

    CASE sy-subrc.
      WHEN 0 .
        LOOP AT std\_text\_raw ASSIGNING <std\_text\_line>.

          CLEAR:
              line\_cluster,
              counter\_lines,
              std\_text\_lines[],
              lines\_cluster[].

          MOVE-CORRESPONDING <std\_text\_line> TO line\_cluster.
          APPEND line\_cluster TO lines\_cluster.

          IMPORT tline = std\_text\_lines FROM INTERNAL TABLE lines\_cluster.

          IF sy-subrc = 0.

            LOOP AT std\_text\_lines ASSIGNING <std\_text\_line\_cluster>.

              counter\_lines = counter\_lines + 1.

              CLEAR data\_line.
              data\_line-text\_obj = <std\_text\_line>-tdobject.
              data\_line-text\_name = <std\_text\_line>-tdname.
              data\_line-text\_id = <std\_text\_line>-tdid.
              data\_line-num\_rec = counter\_lines.
              data\_line-line = <std\_text\_line\_cluster>-tdline.

              APPEND data\_line TO e\_t\_data.

              if counter\_lines >= 999.
                EXIT.
              endif.

            ENDLOOP. " std\_text\_lines

          ENDIF.

        ENDLOOP. " std\_text\_raw

        counter\_datapakage = counter\_datapakage + 1.

      WHEN OTHERS.
        CLOSE CURSOR db\_cursor.
        RAISE no\_more\_data.
    ENDCASE.

  ENDIF.              "Initialization mode or data extraction ?

ENDFUNCTION.

## Листинг 2. Функциональный модуль ZBI\_REMOVE\_INVALID\_CHARACTERS

FUNCTION ZBI\_REMOVE\_INVALID\_CHARACTERS.
*\*"----------------------------------------------------------------------*
*\*"  IMPORTING*
*\*"     REFERENCE(I\_ALLOWED\_CHARACTERS) TYPE  RSALLOWCHAR OPTIONAL*
*\*"     REFERENCE(I\_REPLACE\_CHAR) TYPE  CHAR0001 DEFAULT SPACE*
*\*"     REFERENCE(I\_TEXT) TYPE  RSTXTLG*
*\*"  EXPORTING*
*\*"     REFERENCE(E\_RESULT) TYPE  RSTXTLG*
*\*"----------------------------------------------------------------------*

  DATA ALLOWED\_CHARACTERS TYPE RSALLOWCHAR.

  IF I\_ALLOWED\_CHARACTERS IS INITIAL.
    CALL FUNCTION 'RSKC\_ALLOWED\_CHAR\_GET'
      IMPORTING
        E\_ALLOWED\_CHAR = ALLOWED\_CHARACTERS.
  ELSE.
    ALLOWED\_CHARACTERS = I\_ALLOWED\_CHARACTERS.
  ENDIF.

  E\_RESULT = I\_TEXT.

  REPLACE ALL OCCURRENCES
   OF REGEX |[^]| && ALLOWED\_CHARACTERS && |]|
   IN E\_RESULT
   WITH I\_REPLACE\_CHAR RESPECTING CASE.
  CONDENSE E\_RESULT.

ENDFUNCTION.

## Листинг 3. Подпрограмма определения значения признака ZFKEY\_01

DATA: error TYPE REF TO cx\_root.

TRY .
CASE SOURCE\_FIELDS-text\_obj.
  WHEN 'BELEG'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 4 ).
  WHEN 'DOC\_ITEM'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 4 ).
  WHEN 'EINE' .
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN 'LFB1'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN 'EKKO'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN 'EKPO'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN 'BUT000'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN 'MATERIAL'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 18 ).
  WHEN 'VBBK'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN 'VBBP'.
 RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 0 len  = 10 ).
  WHEN OTHERS.

monitor\_rec-msgid = 'RSRMON'.
monitor\_rec-msgty = 'W'.
monitor\_rec-msgno = '000'.
monitor\_rec-msgv2 = |Text object is not defined: | && SOURCE\_FIELDS-text\_obj.
monitor\_rec-msgv3 = |Record skipped|.
APPEND monitor\_rec TO MONITOR.

RAISE EXCEPTION TYPE CX\_RSROUT\_SKIP\_RECORD.

ENDCASE.

  CATCH cx\_sy\_range\_out\_of\_bounds INTO error.

    monitor\_rec-msgid = 'RSRMON'.
    monitor\_rec-msgty = 'W'.
    monitor\_rec-msgno = '000'.
    monitor\_rec-msgv1 =  error->get\_text( ).
    monitor\_rec-msgv2 = |Field: | && SOURCE\_FIELDS-text\_obj.
    monitor\_rec-msgv3 = |Value: | && SOURCE\_FIELDS-text\_name.
    monitor\_rec-msgv4 = |Record No: | && SOURCE\_FIELDS-RECORD.
    APPEND monitor\_rec TO MONITOR.

    RAISE EXCEPTION TYPE CX\_RSROUT\_SKIP\_RECORD.

ENDTRY.

## Листинг 4. Подпрограмма определения значения признака ZFKEY\_02

DATA: error TYPE REF TO cx\_root.

    TRY .

    CASE SOURCE\_FIELDS-text\_obj.
      WHEN 'BELEG'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 4 len = 10 ).
      WHEN 'DOC\_ITEM'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 4 len = 10 ).
      WHEN 'EINE'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 10 len = 4 ).
      WHEN 'LFB1'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 10 len = 4 ).
      WHEN 'EKPO'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 10 len = 5 ).
      WHEN 'VBBP'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 10 len = 6 ).
      WHEN OTHERS.
        RESULT = ''.
    ENDCASE.

  CATCH cx\_sy\_range\_out\_of\_bounds INTO error.

    monitor\_rec-msgid = 'RSRMON'.
    monitor\_rec-msgty = 'W'.
    monitor\_rec-msgno = '000'.
    monitor\_rec-msgv1 =  error->get\_text( ).
    monitor\_rec-msgv2 = |Field: | && SOURCE\_FIELDS-text\_obj.
    monitor\_rec-msgv3 = |Value: | && SOURCE\_FIELDS-text\_name.
    monitor\_rec-msgv4 = |Record No: | && SOURCE\_FIELDS-RECORD.
    APPEND monitor\_rec TO MONITOR.

    RAISE EXCEPTION TYPE CX\_RSROUT\_SKIP\_RECORD.

ENDTRY.

## Листинг 5. Подпрограмма определения значения признака ZFKEY\_03

DATA: error TYPE REF TO cx\_root.

    TRY .

    CASE SOURCE\_FIELDS-text\_obj.
      WHEN 'BELEG'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 14 len = 4 ).
      WHEN 'DOC\_ITEM'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 14 len = 4 ).
      WHEN 'EINE'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 14 len = 1 ).
      WHEN OTHERS.
        RESULT = ''.
    ENDCASE.

  CATCH cx\_sy\_range\_out\_of\_bounds INTO error.

    monitor\_rec-msgid = 'RSRMON'.
    monitor\_rec-msgty = 'W'.
    monitor\_rec-msgno = '000'.
    monitor\_rec-msgv1 =  error->get\_text( ).
    monitor\_rec-msgv2 = |Field: | && SOURCE\_FIELDS-text\_obj.
    monitor\_rec-msgv3 = |Value: | && SOURCE\_FIELDS-text\_name.
    monitor\_rec-msgv4 = |Record No: | && SOURCE\_FIELDS-RECORD.
    APPEND monitor\_rec TO MONITOR.

    RAISE EXCEPTION TYPE CX\_RSROUT\_SKIP\_RECORD.

ENDTRY.

## Листинг 6. Подпрограмма определения значения признака ZFKEY\_04

DATA: error TYPE REF TO cx\_root.

    TRY .

    CASE SOURCE\_FIELDS-text\_obj.
      WHEN 'BELEG'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 14 len = 4 ).
      WHEN 'DOC\_ITEM'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 14 len = 4 ).
      WHEN 'EINE'.
        RESULT = substring( val = SOURCE\_FIELDS-text\_name off = 14 len = 1 ).
      WHEN OTHERS.
        RESULT = ''.
    ENDCASE.

  CATCH cx\_sy\_range\_out\_of\_bounds INTO error.

    monitor\_rec-msgid = 'RSRMON'.
    monitor\_rec-msgty = 'W'.
    monitor\_rec-msgno = '000'.
    monitor\_rec-msgv1 =  error->get\_text( ).
    monitor\_rec-msgv2 = |Field: | && SOURCE\_FIELDS-text\_obj.
    monitor\_rec-msgv3 = |Value: | && SOURCE\_FIELDS-text\_name.
    monitor\_rec-msgv4 = |Record No: | && SOURCE\_FIELDS-RECORD.
    APPEND monitor\_rec TO MONITOR.

    RAISE EXCEPTION TYPE CX\_RSROUT\_SKIP\_RECORD.

ENDTRY.

## Листинг 7. Программа создания документов ZBI\_UPLOAD\_DOCUMENTS\_MAST

REPORT zbi\_upload\_documents\_mast.

*\*----------------------------------------------------------------------\**
*\*  Text symbols*
*\*----------------------------------------------------------------------\**
*\*KEY  Key components*
*\*OBJ  Documents*
*\*TXT  Text Object*

*\*----------------------------------------------------------------------\**
*\*  Selection Texts*
*\*----------------------------------------------------------------------\**
*\*CHARSET  Charset*
*\*DOC\_TYPE  Document Type*
*\*INFO\_OBJ  InfoObject*
*\*KEY\_01  Key 1*
*\*KEY\_02  Key 2*
*\*KEY\_03  Key 3*
*\*KEY\_04  Key 4*
*\*POS\_01  Pos. 1*
*\*POS\_02  Pos. 2*
*\*POS\_03  Pos. 3*
*\*POS\_04  Pos. 4*
*\*TEXT\_ID  Text ID*
*\*TEXT\_OBJ  Text Object*

INCLUDE zbi\_upload\_documents\_mast\_data.
INCLUDE zbi\_upload\_documents\_mast\_sscr.

INITIALIZATION.
  PERFORM set\_restrict\_sel\_options.

START-OF-SELECTION.

  PERFORM check\_infoobject\_is\_exist.
  PERFORM set\_replace\_pattern.
  PERFORM select\_contents\_documents.

  LOOP AT contents\_documents ASSIGNING <doc\_line>.

    PERFORM check\_is\_new\_doc.

    IF is\_new\_doc = abap\_true.

      PERFORM set\_doc\_preference.
      PERFORM set\_doc\_content.
      PERFORM create\_doc.
      PERFORM print\_message.
      PERFORM clear\_long\_line.

    ENDIF. *" is\_new\_doc*

    PERFORM set\_long\_line.
    PERFORM save\_current\_doc\_key.

  ENDLOOP.

FORM create\_doc.

  CALL FUNCTION 'RSOD\_DOC\_MAST\_CHANGE'
    EXPORTING
      i\_chanm                = infoobject\_name
      i\_chavl                = infoobject\_value
      i\_doc\_type             = doc\_type
      i\_description          = doc\_description
      i\_name                 = doc\_name
      i\_overwrite\_mode       = if\_rsod\_const=>mode\_replace\_phio
      i\_with\_content         = rs\_c\_true
      i\_s\_content\_info       = doc\_info
    IMPORTING
      e\_s\_return             = message
    TABLES
      i\_t\_file\_content\_ascii = doc\_content.

ENDFORM.                    *"create\_doc*

FORM print\_message.

  IF message-type IS INITIAL.
    WRITE: /  infoobject\_name  QUICKINFO 'Infoobject' ##no\_text,
              infoobject\_value QUICKINFO 'Value'      ##no\_text,
              doc\_description  QUICKINFO 'Desc'       ##no\_text,
              doc\_name         QUICKINFO 'Desc'       ##no\_text,
              'Ok'             QUICKINFO 'Type'       ##no\_text,
              space            QUICKINFO  'Id'        ##no\_text,
              space            QUICKINFO  'Number'    ##no\_text,
              space            QUICKINFO  'Message'   ##no\_text.
  ELSE.
    WRITE: /  infoobject\_name  QUICKINFO 'Infoobject' ##no\_text,
              infoobject\_value QUICKINFO 'Value'      ##no\_text,
              doc\_description  QUICKINFO 'Desc'       ##no\_text,
              doc\_name         QUICKINFO 'Desc'       ##no\_text,
              message-type     QUICKINFO 'Type'       ##no\_text,
              message-id       QUICKINFO  'Id'        ##no\_text,
              message-number   QUICKINFO  'Number'    ##no\_text,
              message-message  QUICKINFO  'Message'   ##no\_text.
  ENDIF.

ENDFORM.                    *"print\_massage*

FORM check\_is\_new\_doc.
  is\_new\_doc = boolc( <doc\_line>-/bic/znum\_rec <= doc\_key-/bic/znum\_rec  ).
ENDFORM.                    *"check\_is\_is\_new\_line*

FORM clear\_long\_line.
  CLEAR lines.
ENDFORM.                    *"clear\_long\_line*

FORM save\_current\_doc\_key.
  MOVE-CORRESPONDING <doc\_line> TO doc\_key.
ENDFORM.                    *"save\_current\_doc\_key*

FORM set\_long\_line.

  lines = lines &&
        <doc\_line>-/bic/zline\_01 && | | &&
        <doc\_line>-/bic/zline\_02 && | | &&
        <doc\_line>-/bic/zline\_03 && | |.

ENDFORM.                    *"set\_long\_line*

FORM set\_doc\_preference.

  infoobject\_name = info\_obj.

  PERFORM set\_mimetype.
  PERFORM set\_doc\_name.
  PERFORM set\_doc\_description.
  PERFORM set\_infoobject\_value.

ENDFORM.                    *"set\_doc\_preference*

FORM set\_mimetype.

  IF charset IS NOT INITIAL.
    doc\_info-mimetype = mimetype\_text && |; charset=| && charset.
  ELSE.
    doc\_info-mimetype = mimetype\_text.
  ENDIF.

ENDFORM.                    *"set\_mimetype*

FORM set\_doc\_content.

  DATA:
    file\_info TYPE sdokfilaci.

  CLEAR file\_info.
  MOVE-CORRESPONDING doc\_info TO file\_info.
  lines = condense( val = lines del = | | ).

  cl\_rsod\_utilities=>string\_to\_content\_table(
                      EXPORTING i\_string  = lines
                      IMPORTING e\_t\_table = doc\_content
                      CHANGING c\_s\_file\_info  = file\_info ).

ENDFORM.                    *"set\_doc\_content*

FORM set\_doc\_name.
  doc\_name = infoobject\_name && text\_obj && text\_id &&
             doc\_key-/bic/zfkey\_01 &&
             doc\_key-/bic/zfkey\_02 &&
             doc\_key-/bic/zfkey\_03 &&
             doc\_key-/bic/zfkey\_04.
ENDFORM.                    *"set\_doc\_name*

FORM set\_doc\_description.

  IF doc\_type IS INITIAL.
    doc\_description = text\_obj && | | && text\_id && | | &&
                      doc\_key-/bic/zfkey\_01 &&
                      doc\_key-/bic/zfkey\_02 &&
                      doc\_key-/bic/zfkey\_03 &&
                      doc\_key-/bic/zfkey\_04.
  ELSE.
    doc\_description = doc\_type && | | &&
                      doc\_key-/bic/zfkey\_01 &&
                      doc\_key-/bic/zfkey\_02 &&
                      doc\_key-/bic/zfkey\_03 &&
                      doc\_key-/bic/zfkey\_04.
  ENDIF.

ENDFORM.                    *"set\_doc\_description*

FORM set\_restrict\_sel\_options.

  DATA:
    restrict TYPE sscr\_restrict,
    list\_options TYPE sscr\_opt\_list,
    associating  TYPE sscr\_ass.

  MOVE 'ALLOWED' TO list\_options-name.
  MOVE 'X' TO: list\_options-options-bt,
               list\_options-options-eq,
               list\_options-options-ge,
               list\_options-options-gt,
               list\_options-options-le,
               list\_options-options-lt.
  APPEND list\_options TO restrict-opt\_list\_tab.

  CLEAR associating.
  MOVE: 'B'         TO associating-kind,
        'SS\_KEYS'   TO associating-name,
        'I'         TO associating-sg\_main,
        ''          TO associating-sg\_addy,
        'ALLOWED'   TO associating-op\_main.
  APPEND associating TO restrict-ass\_tab.

  CALL FUNCTION 'SELECT\_OPTIONS\_RESTRICT'
    EXPORTING
      restriction            = restrict
    EXCEPTIONS
      too\_late               = 1
      repeated               = 2
      selopt\_without\_options = 3
      selopt\_without\_signs   = 4
      invalid\_sign           = 5
      empty\_option\_list      = 6
      invalid\_kind           = 7
      repeated\_kind\_a        = 8
      OTHERS                 = 9.

  IF sy-subrc <> 0.
    MESSAGE ID sy-msgid TYPE 'W' NUMBER sy-msgno
            WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
  ENDIF.

ENDFORM.                    *"set\_restrict*

FORM check\_infoobject\_is\_exist.

  DATA: tlogos TYPE rso\_th\_tlogo,
        tlogo  TYPE rso\_s\_tlogo,
        tlogo\_objnm TYPE rschanm.

  tlogo\_objnm = info\_obj.
  tlogo-objnm = tlogo\_objnm.
  tlogo-tlogo = rs\_c\_tlogo-infoobject.
  INSERT tlogo INTO TABLE tlogos.

  TRY.
      cl\_rsd\_iobj=>get\_info\_current(
        EXPORTING i\_th\_tlogo = tlogos
                  i\_iobjnm   = tlogo\_objnm ).
    CATCH cx\_rsd\_iobj\_not\_found.
      MESSAGE w106(r7) WITH infoobject\_name.
      LEAVE PROGRAM.
  ENDTRY.

ENDFORM.                    *"check\_infoobject\_is\_exist*

FORM select\_contents\_documents.

  SELECT \*
    FROM  /bic/azca\_o0100
    INTO  CORRESPONDING FIELDS OF TABLE contents\_documents
    WHERE      /bic/ztext\_obj  = text\_obj
           AND /bic/ztext\_id   = text\_id
           AND /bic/zfkey\_01   IN key\_01
           AND /bic/zfkey\_02   IN key\_02
           AND /bic/zfkey\_03   IN key\_03
           AND /bic/zfkey\_04   IN key\_04.

ENDFORM.                    *"select\_contents\_documents*

FORM set\_infoobject\_value.

  DATA:
    concatenate\_keys TYPE string,
    replace\_value TYPE string.

  concatenate\_keys = |(| && doc\_key-/bic/zfkey\_01 && |)\_| &&
                     |(| && doc\_key-/bic/zfkey\_02 && |)\_| &&
                     |(| && doc\_key-/bic/zfkey\_03 && |)\_| &&
                     |(| && doc\_key-/bic/zfkey\_04 && |)| .

  replace\_value =    doc\_key-/bic/zfkey\_01 && |\_| &&
                     doc\_key-/bic/zfkey\_02 && |\_| &&
                     doc\_key-/bic/zfkey\_03 && |\_| &&
                     doc\_key-/bic/zfkey\_04 .

  infoobject\_value = replace( val   = replace\_value
                              regex = concatenate\_keys
                              with  = replace\_pattern  ).

ENDFORM.                    *"set\_infoobject\_value*

FORM set\_replace\_pattern.

  DATA: concatenate\_pos TYPE string.

  CLEAR concatenate\_pos.

  IF pos\_01 IS NOT INITIAL.
    concatenate\_pos = concatenate\_pos && |$| && pos\_01.
  ENDIF.
  IF pos\_02 IS NOT INITIAL.
    concatenate\_pos = concatenate\_pos && |$| && pos\_02.
  ENDIF.
  IF pos\_03 IS NOT INITIAL.
    concatenate\_pos = concatenate\_pos && |$| && pos\_03.
  ENDIF.
  IF pos\_04 IS NOT INITIAL.
    concatenate\_pos = concatenate\_pos && |$| && pos\_04.
  ENDIF.

  replace\_pattern = concatenate\_pos.

  IF strlen( replace\_pattern ) = 0 .
    MESSAGE 'Key positions are not set' TYPE 'E' ##no\_text.
    LEAVE PROGRAM.
  ENDIF.

ENDFORM.                    *"set\_replace\_pattern*

*\*&---------------------------------------------------------------------\**
*\*&  Include           ZBI\_UPLOAD\_DOCUMENTS\_MAST\_DATA*
*\*&---------------------------------------------------------------------\**

TYPE-POOLS sscr.

TABLES:
    /bic/azca\_o0100.

TYPES:
      BEGIN OF ty\_doc\_line,
          /bic/zfkey\_01 TYPE /bic/oizfkey\_01,
          /bic/zfkey\_02 TYPE /bic/oizfkey\_02,
          /bic/zfkey\_03 TYPE /bic/oizfkey\_03,
          /bic/zfkey\_04 TYPE /bic/oizfkey\_04,
          /bic/znum\_rec TYPE /bic/oiznum\_rec,
          /bic/zline\_01 TYPE /bic/oizline\_01,
          /bic/zline\_02 TYPE /bic/oizline\_02,
          /bic/zline\_03 TYPE /bic/oizline\_03,
      END OF ty\_doc\_line,

      BEGIN OF ty\_doc\_key,
          /bic/zfkey\_01 TYPE /bic/oizfkey\_01,
          /bic/zfkey\_02 TYPE /bic/oizfkey\_02,
          /bic/zfkey\_03 TYPE /bic/oizfkey\_03,
          /bic/zfkey\_04 TYPE /bic/oizfkey\_04,
          /bic/znum\_rec TYPE /bic/oiznum\_rec,
      END OF ty\_doc\_key.

CONSTANTS:
    mimetype\_text TYPE w3conttype VALUE 'text/plain'.

FIELD-SYMBOLS:
    <doc\_line> TYPE ty\_doc\_line ##needed.

DATA:
    infoobject TYPE rsiobjnm ##needed,
    infoobject\_name TYPE rschanm ##needed,
    infoobject\_value TYPE rschavl ##needed,

    lines TYPE string ##needed,
    replace\_pattern TYPE string ##needed,
    is\_new\_doc TYPE abap\_bool ##needed,
    message TYPE bapiret2 ##needed,

    doc\_description TYPE sdok\_descr ##needed,
    doc\_name TYPE skwf\_urlp ##needed,
    doc\_info TYPE rsod\_s\_content\_info ##needed,
    doc\_key TYPE ty\_doc\_key ##needed.

DATA:
    contents\_documents TYPE SORTED TABLE OF ty\_doc\_line ##needed
                        WITH NON-UNIQUE KEY /bic/zfkey\_01
                                            /bic/zfkey\_02
                                            /bic/zfkey\_03
                                            /bic/zfkey\_04,
    doc\_content TYPE TABLE OF sdokcntasc ##needed.

*\*&---------------------------------------------------------------------\**
*\*&  Include           ZBI\_UPLOAD\_DOCUMENTS\_MAST\_SSCR*
*\*&---------------------------------------------------------------------\**

SELECTION-SCREEN BEGIN OF BLOCK ss\_text\_obj WITH FRAME TITLE text-txt.
PARAMETERS:
  text\_obj TYPE /bic/azca\_o0100-/bic/ztext\_obj OBLIGATORY MATCHCODE OBJECT h\_ttxob,
  text\_id TYPE /bic/azca\_o0100-/bic/ztext\_id OBLIGATORY.
SELECTION-SCREEN END OF BLOCK ss\_text\_obj.

SELECTION-SCREEN BEGIN OF BLOCK ss\_keys WITH FRAME TITLE text-key.

SELECTION-SCREEN BEGIN OF LINE.
PARAMETERS pos\_01 TYPE n LENGTH 1.
SELECT-OPTIONS key\_01 FOR /bic/azca\_o0100-/bic/zfkey\_01.
SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN BEGIN OF LINE.
PARAMETERS pos\_02 TYPE n LENGTH 1.
SELECT-OPTIONS key\_02 FOR /bic/azca\_o0100-/bic/zfkey\_02.
SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN BEGIN OF LINE.
PARAMETERS pos\_03 TYPE n LENGTH 1.
SELECT-OPTIONS key\_03 FOR /bic/azca\_o0100-/bic/zfkey\_03.
SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN BEGIN OF LINE.
PARAMETERS pos\_04 TYPE n LENGTH 1.
SELECT-OPTIONS key\_04 FOR /bic/azca\_o0100-/bic/zfkey\_04.
SELECTION-SCREEN END OF LINE.

SELECTION-SCREEN END OF BLOCK ss\_keys.

SELECTION-SCREEN BEGIN OF BLOCK ss\_iobj WITH FRAME TITLE text-obj.
PARAMETERS:
  info\_obj TYPE rsdchabas-chabasnm OBLIGATORY MATCHCODE OBJECT rsod\_chanm MEMORY ID rsc,
  doc\_type TYPE rsodwwwdoctype,
  charset TYPE w3conttype DEFAULT  'cp1251'.
SELECTION-SCREEN END OF BLOCK ss\_iobj.

## Листинг 8. Пользовательская функция CommentValue

Public Function CommentValue**(**Cell As Range**,** Optional Delimiter As String **=** vbNewLine**)** As String

 CommentValue **=** vbNullString

 If Not Cell.Comment Is Nothing Then

 CommentValue **=** Replace**(**Cell.Comment.Text**,** vbLf**,** Delimiter**)**

 End If

End Functio