

```

*&-----*
*& Report  ZB3_ALE_IDOC
*&
*&-----*
*&
*&
*&-----*

```

REPORT ZB3\_ALE\_IDOC NO STANDARD PAGE HEADING LINE-COUNT 27 LINE-SIZE 225.

```

TYPES : BEGIN OF TY_MARA,
        MATNR TYPE MATNR,
        MTART TYPE MTART,
        MBRSH TYPE MBRSH,
        MEINS TYPE MEINS,
        END OF TY_MARA,
        BEGIN OF TY_MAKT,
        MATNR TYPE MATNR,
        SPRAS TYPE SPRAS,
        MAKTX TYPE MAKTX,
        END OF TY_MAKT,
        BEGIN OF TY_MARC,
        MATNR TYPE MATNR,
        WERKS TYPE WERKS_D,
        MINBE TYPE MINBE,
        EISBE TYPE EISBE,
        END OF TY_MARC,
        BEGIN OF TY_MARD,
        MATNR TYPE MATNR,
        WERKS TYPE WERKS_D,
        LGORT TYPE LGORT_D,
        LABST TYPE LABST,
        END OF TY_MARD.

```

```

TYPES : BEGIN OF TY_FILE,
        LINE(500) TYPE C,
*        LINE1(500) TYPE C,
*        LINE2(500) TYPE C,
*        LINE3(500) TYPE C,
        END OF TY_FILE.

```

```

TYPES : TT_MARA TYPE STANDARD TABLE OF TY_MARA INITIAL SIZE 1,
        TT_MAKT TYPE STANDARD TABLE OF TY_MAKT INITIAL SIZE 1,
        TT_MARC TYPE STANDARD TABLE OF TY_MARC INITIAL SIZE 1,
        TT_MARD TYPE STANDARD TABLE OF TY_MARD INITIAL SIZE 1,
        TT_FILE TYPE STANDARD TABLE OF TY_FILE INITIAL SIZE 1.

```

```

DATA : T_MARA TYPE TT_MARA,
        T_MAKT TYPE TT_MAKT,
        T_MARC TYPE TT_MARC,
        T_MARD TYPE TT_MARD,
        T_FILE TYPE TT_FILE,
        W_MARA TYPE TY_MARA,
        W_MAKT TYPE TY_MAKT,
        W_MARC TYPE TY_MARC,
        W_MARD TYPE TY_MARD,
        W_FILE TYPE TY_FILE.

```

```

DATA : LV_MATNR TYPE MATNR,
        LV_MTART TYPE MTART,
        LV_MBRSH TYPE MBRSH,
        LV_MEINS TYPE MEINS.

```

```

DATA : LV_TEXT TYPE STRING,
       LV_TEXT1 TYPE STRING,
       LV_TEXT2 TYPE STRING,
       LV_TEXT3 TYPE STRING.

DATA : LV_MINBE TYPE STRING,
       LV_EISBE TYPE STRING,
       LV_LABST TYPE STRING.

DATA : V_MATNR TYPE MATNR,
       V_WERKS TYPE WERKS_D,
       V_MTART TYPE MTART,
       V_MBRSH TYPE MBRSH.

PARAMETERS P_PATH TYPE STRING.

TABLES : MARA,MARC.

SELECTION-SCREEN BEGIN OF BLOCK B1 WITH FRAME TITLE TEXT-001.
SELECT-OPTIONS : S_MATNR FOR MARA-MATNR,
                 S_WERKS FOR MARC-WERKS,
                 S_MTART FOR MARA-MTART,
                 S_MBRSH FOR MARA-MBRSH.

SELECTION-SCREEN END OF BLOCK B1.

AT SELECTION-SCREEN.
  IF S_MATNR IS NOT INITIAL.
    SELECT SINGLE MATNR INTO V_MATNR FROM MARA WHERE MATNR IN S_MATNR.
    IF SY-SUBRC <> 0.
      MESSAGE TEXT-008 TYPE 'E'.
    ENDIF.
  ENDIF.

  IF S_WERKS IS NOT INITIAL.
    SELECT SINGLE WERKS INTO V_WERKS FROM MARC WHERE WERKS IN S_WERKS.
    IF SY-SUBRC <> 0.
      MESSAGE TEXT-009 TYPE 'E'.
    ENDIF.
  ENDIF.

  IF S_MTART IS NOT INITIAL.
    SELECT SINGLE MTART INTO V_MTART FROM T134 WHERE MTART IN S_MTART.
    IF SY-SUBRC <> 0.
      MESSAGE TEXT-010 TYPE 'E'.
    ENDIF.
  ENDIF.

  IF S_MBRSH IS NOT INITIAL.
    SELECT SINGLE MBRSH INTO V_MBRSH FROM T137 WHERE MBRSH IN S_MBRSH.
    IF SY-SUBRC <> 0.
      MESSAGE TEXT-011 TYPE 'E'.
    ENDIF.
  ENDIF.

  ENDIF.

AT SELECTION-SCREEN ON VALUE-REQUEST FOR P_PATH.
  CALL FUNCTION 'GUI_FILE_SAVE_DIALOG'
  IMPORTING
    FULLPATH = P_PATH.

START-OF-SELECTION.

  SELECT MATNR MTART MBRSH MEINS INTO TABLE T_MARA FROM MARA
    WHERE MATNR IN S_MATNR AND MTART IN S_MTART AND MBRSH IN S_MBRSH.
  IF SY-SUBRC <> 0.
    MESSAGE TEXT-006 TYPE 'I'.
    LEAVE LIST-PROCESSING.
  ENDIF.

```

```

IF T_MARA IS NOT INITIAL.
  SELECT MATNR SPRAS MAKTX INTO TABLE T_MAKT FROM MAKT FOR ALL ENTRIES IN T_MARA
    WHERE MATNR = T_MARA-MATNR.
  IF SY-SUBRC = 0.
    SORT T_MAKT BY MATNR.
  ENDIF.
  SELECT MATNR WERKS MINBE EISBE INTO TABLE T_MARC FROM MARC FOR ALL ENTRIES IN T_MARA
    WHERE MATNR = T_MARA-MATNR AND WERKS IN S_WERKS.
  IF SY-SUBRC <> 0 AND S_WERKS IS NOT INITIAL.
    MESSAGE TEXT-007 TYPE 'I'.
    LEAVE LIST-PROCESSING.
  ENDIF.
ENDIF.

IF T_MARC IS NOT INITIAL.
  SELECT MATNR WERKS LGORT LABST INTO TABLE T_MARD FROM MARD FOR ALL ENTRIES IN T_MARC
    WHERE MATNR = T_MARC-MATNR AND WERKS = T_MARC-WERKS.
  IF SY-SUBRC = 0.
    SORT T_MARD BY MATNR.
  ENDIF.
ENDIF.

END-OF-SELECTION.

```

```

LV_TEXT = TEXT-002.
LV_TEXT1 = TEXT-003.
LV_TEXT2 = TEXT-004.
LV_TEXT3 = TEXT-005.
CLEAR W_MARA.

```

```

LOOP AT T_MARA INTO W_MARA.

```

```

  LV_MATNR = W_MARA-MATNR.
  LV_MTART = W_MARA-MTART.
  LV_MBRSH = W_MARA-MBRSH.
  LV_MEINS = W_MARA-MEINS.

```

```

  AT NEW MATNR.
    CONCATENATE LV_TEXT LV_MATNR LV_MTART LV_MBRSH LV_MEINS INTO W_FILE-LINE.
    APPEND W_FILE TO T_FILE.
    CLEAR W_FILE.

```

```

  ENDAT.
  CLEAR W_MAKT.
  LOOP AT T_MAKT INTO W_MAKT WHERE MATNR = W_MARA-MATNR.

```

```

    CONCATENATE LV_TEXT1 W_MAKT-SPRAS W_MAKT-MAKTX INTO W_FILE-LINE.
    APPEND W_FILE TO T_FILE.
    CLEAR W_FILE.

```

```

  ENDLOOP.
  CLEAR W_MARC.
  LOOP AT T_MARC INTO W_MARC WHERE MATNR = W_MARA-MATNR.
    LV_MINBE = W_MARC-MINBE.
    LV_EISBE = W_MARC-EISBE.

```

```

    CONCATENATE LV_TEXT2 W_MARC-WERKS LV_MINBE LV_EISBE INTO W_FILE-LINE.
    APPEND W_FILE TO T_FILE.
    CLEAR W_FILE.

```

```

ENDLOOP.
CLEAR W_MARD.
LOOP AT T_MARD INTO W_MARD WHERE MATNR = W_MARC-MATNR.
    LV_LABST = W_MARD-LABST.
    CONCATENATE LV_TEXT3 W_MARD-LGORT LV_LABST INTO W_FILE-LINE.
    APPEND W_FILE TO T_FILE.
    CLEAR W_FILE.
ENDLOOP.

ENDLOOP.

CALL FUNCTION 'GUI_DOWNLOAD'
EXPORTING
    FILENAME                = P_PATH
TABLES
    DATA_TAB                = T_FILE
EXCEPTIONS
    FILE_WRITE_ERROR        = 1
    NO_BATCH                 = 2
    GUI_REFUSE_FILETRANSFER = 3
    INVALID_TYPE             = 4
    NO_AUTHORITY             = 5
    UNKNOWN_ERROR            = 6
    HEADER_NOT_ALLOWED       = 7
    SEPARATOR_NOT_ALLOWED    = 8
    FILESIZE_NOT_ALLOWED     = 9
    HEADER_TOO_LONG          = 10
    DP_ERROR_CREATE          = 11
    DP_ERROR_SEND            = 12
    DP_ERROR_WRITE           = 13
    UNKNOWN_DP_ERROR         = 14
    ACCESS_DENIED            = 15
    DP_OUT_OF_MEMORY         = 16
    DISK_FULL                = 17
    DP_TIMEOUT               = 18
    FILE_NOT_FOUND           = 19
    DATAPROVIDER_EXCEPTION   = 20
    CONTROL_FLUSH_ERROR      = 21
    OTHERS                   = 22.
IF SY-SUBRC <> 0.
* Implement suitable error handling here
ENDIF.

```