

```

*/-----}
*/ This file is part of ZTABLE_MOVER. |
*/ Copyright (c) 2007, Consnet (pty) Ltd |
*/ All rights reserved. |
*/ |
*/ Redistribution and use in source and binary forms, with or without |
*/ modification, are permitted provided that the following conditions |
*/ are met: |
*/ 1. Redistributions of source code must retain the above copyright |
*/ notice, this list of conditions and the following disclaimer. |
*/ 2. Redistributions in binary form must reproduce the above copyright |
*/ notice, this list of conditions and the following disclaimer in |
*/ the documentation and/or other materials provided with the |
*/ distribution. |
*/ 3. Neither the name of the Consnet (pty) Ltd nor the names of its |
*/ contributors may be used to endorse or promote products derived |
*/ from this software without specific prior written permission. |
*/ THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS |
*/ "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT |
*/ LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS |
*/ FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE |
*/ COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, |
*/ INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, |
*/ BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; |
*/ LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER |
*/ CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT |
*/ LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN |
*/ ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE |
*/ POSSIBILITY OF SUCH DAMAGE. |
*/-----/
REPORT ZTABLE_MOVER.

type-pools: abap.
tables: rsrdl.

types: begin of t_dnl,
       tabname      like rsrdl-tbma_val,
       content       type text4096,
       end of t_dnl.

data: lt_dnl type table of t_dnl,
      ls_dnl type t_dnl.

data: lr_table type ref to data,
      lr_line  type ref to data.

field-symbols: <lt_table> type standard table,
               <ls_table> type any.

data: lt_string type table of string,
      ls_string type string.

field-symbols: <ls_line> type any,
               <lv_content> type any,
               <lv_field> type any.

data: lv_1024 type text1024.

data: sep type abap_char1,
      c_sep type abap_char1.

data: lt_files type filetable,
      ls_files type file_table,
      lv_action type i,
      lv_answer type c,
      lv_rc type sysubrc,
      lv_name type string.

data: lv_tab like rsrdl-tbma_val.
data: lt_tables type table of dd02l,
      ls_tables type dd02l.

*****
* Selection Screen *
*-----*
select-options: so_tab for rsrdl-tbma_val.

```

```

parameters: p_up type flag as checkbox,
            p_down type flag as checkbox,
            p_file(300) type c modif id did obligatory.

*****
* AT SELECTION SCREEN *
*-----*
at selection-screen on value-request for p_file.
  call method cl_gui_frontend_services=>file_open_dialog
    exporting
      multiselection = abap_false
      file_filter = '*.txt'
      default_extension = 'txt'
    changing
      file_table = lt_files
      rc = lv_rc
      user_action = lv_action.
  read table lt_files into ls_files index 1.
  p_file = ls_files.

  clear: lt_files, ls_files.

*****
* START-OF-SELECTION *
*-----*
start-of-selection.

  c_sep = cl_abap_char_utilities=>horizontal_tab.

  if p_up = 'X'.
    call function 'POPUP_TO_CONFIRM_STEP'
      exporting
        defaultoption = 'Y'
        textline1 = 'STOP! Are you sure you want to Continue?'
        textline2 = 'Uploading these tables can Destroy your System'
        titel = 'Confirm'
      importing
        answer = lv_answer.

    if lv_answer = 'J'.

      lv_name = p_file.
      call method cl_gui_frontend_services=>gui_upload
        exporting
          filename = lv_name
          filetype = 'ASC'
          has_field_separator = ''
        changing
          data_tab = lt_dnl
        exceptions
          file_open_error = 1
          file_read_error = 2
          no_batch = 3
          gui_refuse_filetransfer = 4
          invalid_type = 5
          no_authority = 6
          unknown_error = 7
          bad_data_format = 8
          header_not_allowed = 9
          separator_not_allowed = 10
          header_too_long = 11
          unknown_dp_error = 12
          access_denied = 13
          dp_out_of_memory = 14
          disk_full = 15
          dp_timeout = 16
          not_supported_by_gui = 17
          error_no_gui = 18
          others = 19.
      if sy-subrc = 0.
        sort lt_dnl by tabname.
        loop at lt_dnl into ls_dnl.
          at new tabname.
            lv_tab = ls_dnl-tabname.
            create data lr_table type standard table of (lv_tab)

```

```

        with non-unique default key.
    assign lr_table->* to <lt_table>.
    create data lr_line type (lv_tab).
    assign lr_line->* to <ls_line>.
endat.

clear <ls_line>.
clear lt_string.
split ls_dnl-content at c_sep into table lt_string.
loop at lt_string into ls_string.
    assign component sy-tabix
        of structure <ls_line> to <lv_field>.
    <lv_field> = ls_string.
endloop.
append <ls_line> to <lt_table>.

at end of tabname.
delete from (lv_tab).
commit work and wait.
insert (lv_tab)
    from table <lt_table>.
commit work and wait.
if sy-subrc = 0.
    message s000(0k) with 'Table successfully uploaded'.
endif.
unassign: <lt_table>, <ls_line>.
free: lr_table, lr_line.
endat.
endloop.

endif.
endif.
endif.

clear: lt_dnl, ls_dnl.

if p_down = 'X'.

select * into table lt_tables
from dd021
where tabname in so_tab
and as4local = 'A'
and tabclass = 'TRANSP'.

loop at lt_tables into ls_tables.
clear ls_dnl.

lv_tab = ls_tables-tabname.

create data lr_table type standard table of (lv_tab)
with non-unique default key.

assign lr_table->* to <lt_table>.

lv_name = p_file.

select * into table <lt_table>
from (lv_tab).
loop at <lt_table> assigning <ls_table>.
ls_dnl-tabname = lv_tab.
clear: ls_dnl-content, sy-subrc, sep.
while sy-subrc = 0.
    assign component sy-index of
        structure <ls_table> to <lv_field>.
    if sy-subrc = 0.
        lv_1024 = <lv_field>.
        concatenate ls_dnl-content sep lv_1024 into ls_dnl-content.
        sep = c_sep.
    endif.
endwhile.

append ls_dnl to lt_dnl.

```

```

        endloop.

endloop.

cl_gui_frontend_services=>gui_download(
  exporting
    filename           = lv_name
    filetype           = 'ASC'
    write_field_separator = ' '
  changing
    data_tab           = lt_dnl
  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
    not_supported_by_gui = 22
    error_no_gui        = 23
    others              = 24
).

if sy-subrc = 0.
  message s000(0k) with 'Table successfully downloaded'.
endif.
endif._

```