\*&---------------------------------------------------------------------\*  
\*& Report  Z31\_PARKLANE\_OUTBOUND  
\*&  The Parklane interface file is created each pay run and placed on the CFS drive for Human Resources  
\*&  so that the data can be loaded to the Parklane system used by EHS  
\*&---------------------------------------------------------------------\*  
\*&  Sept-09-2021: Innitial build.  
\*&  Jan-20-2022:  Final build after SIT1 defect fixes.  
\*&  
\*&---------------------------------------------------------------------\*  
report z31\_parklane\_wsib.  
  
  
include zi31\_data\_defn.  
include zi31\_sel\_screen.  
include zi31\_form\_routines.  
  
initialization.  
  "  concatenate '/usr/sap/' sy-sysid '/DVEBMGS00/work/integrations/outbound/31'  into p\_path.  
  concatenate '/sapmnt/' sy-sysid '/customerdata/INTEGRATIONS/OUTBOUND/31\_WSIB\_EE'  into p\_path. "/sapmnt/P10/customerdata/INTEGRATIONS/OUTBOUND/31\_WSIB\_EE  
  p\_file  = 'PARKLANE.DAT'.  
  
\*at selection-screen on value-request for p\_path.  
\*\* concatenate '/usr/sap/' sy-sysid '/DVEBMGS00/work/integrations/outbound/31'  into p\_file.  
\*  call function '/SAPDMC/LSM\_F4\_SERVER\_FILE'  
\*    exporting  
\*      directory        = p\_path  
\*    importing  
\*      serverfile       = p\_path  
\*    exceptions  
\*      canceled\_by\_user = 1  
\*      others           = 2.  
\*  ""  
at selection-screen output.  
  perform set\_sel\_screen\_properties.  
  
  
  
start-of-selection.  
\* Setup date parameters for the program execution  
  perform date\_selection.  
  
\* Get Employee number.  
get peras.  
  
\* Reset global variables before executing next data set.  
  perform cleanup.  
  
\* Get data for selected employee.  
  perform data\_selection.  
  
end-of-selection.  
  
\* Check for errors in output. - To be updated in future SIT cycle.  
  perform check\_errors.  
  
\* Send file to AL11  
  perform write\_csv.  
  
\* Send error report. Update error report as per spec. - To be updated in future SIT cycle.  
  perform send\_error\_report.  
\* Show output in ALV - This is not required for integration as file is already sent to AL11 directory.  
\* This helps with testing of integrations but should be disabled for large datasets.  
  if gv\_fatal\_error eq abap\_false.  
    perform alv\_report.  
  endif.

\*&---------------------------------------------------------------------\*  
\*&  Include           ZI31\_FORM\_ROUTINES  
\*&  Program:          Z31\_PARKLANE\_OUTBOUND  
\*&---------------------------------------------------------------------\*  
form data\_selection.  
  data lv\_is\_main\_pernr type boole\_d value ''.  
  
\* Check input pernr is main pernr  
  perform check\_main\_pernr changing lv\_is\_main\_pernr.  
  if lv\_is\_main\_pernr eq abap\_true.  
  
  
    perform read\_personid                                   changing gv\_personid.  
    perform read\_infty      using gv\_begda gv\_endda '0000'  changing gta\_pa0000.  
\*  
  
    loop at gta\_pa0000 into gwa\_pa0000.  
  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0001'  changing gta\_pa0001.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0002'  changing gta\_pa0002.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0006'  changing gta\_pa0006.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0007'  changing gta\_pa0007.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0008'  changing gta\_pa0008.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0014'  changing gta\_pa0014.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0463'  changing gta\_pa0463.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0464'  changing gta\_pa0464.  
      perform read\_infty      using gwa\_pa0000-begda gwa\_pa0000-endda '0712'  changing gta\_pa0712.  
  
  
  
\*       Constant Values  
      perform fill\_constant\_values.  
  
  
      if gv\_personid is not initial.  
        condense gv\_personid no-gaps.  
        gwa\_output-empid     =  gv\_personid.  
  
  
        call function 'CONVERSION\_EXIT\_ALPHA\_INPUT'  
          exporting  
            input  = gwa\_output-empid  
          importing  
            output = gwa\_output-empid.  
  
  
      endif.  
  
\*        occflag  
      read table gta\_pa0712 with key subty = 'MAIN' transporting no fields.  
  
      if sy-subrc eq 0. " Main person assignment exists.  
        gwa\_output-occflag     =  '\*'.  
      else.  
        gwa\_output-occflag     =  '  '.  
      endif.  
  
  
  
  
      " Status  
      case gwa\_pa0000-stat2.  
        when '0'.  
          gwa\_output-status = 'W'.  
        when '1'.  
          gwa\_output-status = 'I'.  
        when '2'.  
          gwa\_output-status = 'R'.  
        when '3'.  
          gwa\_output-status = 'A'.  
        when others.  
      endcase.  
  
  
\* empldate  
      perform read\_date\_type  using gwa\_pa0000-begda gwa\_pa0000-endda '01'    changing gv\_entry\_date.  
      gwa\_output-empldate = gv\_entry\_date.  
  
\* yearsexp  
      perform get\_year\_from\_date using gv\_entry\_date changing gwa\_output-yearsexp.  
  
\* ltdstart  
      if gwa\_pa0000-massn eq '03' and gwa\_pa0000-massg eq '05'.  
        gwa\_output-ltdstart = gwa\_pa0000-begda.  
  
      elseif  gwa\_pa0000-massn eq '03' and gwa\_pa0000-massg eq '08'.  
  
\* loastart / end.  
        gwa\_output-loastart   = gwa\_pa0000-begda.  
        gwa\_output-loaend     = gwa\_pa0000-endda.  
      elseif gwa\_pa0000-massn eq '03' and gwa\_pa0000-massg eq '03'.  
  
\* matstart / end  
        gwa\_output-matstart   = gwa\_pa0000-begda.  
        gwa\_output-matend     = gwa\_pa0000-endda.  
  
      endif.  
  
  
      loop at gta\_pa0002 into gwa\_pa0002 where pernr = gwa\_pa0000-pernr and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda .  
\*   Personal details from infoytpe 0002.  
        gwa\_output-sin        =   gwa\_pa0002-perid .  
        gwa\_output-surname    =   gwa\_pa0002-nachn.  
        gwa\_output-language   =   gwa\_pa0002-sprsl.  
        gwa\_output-dob        =   gwa\_pa0002-gbdat .  
        gwa\_output-givename   =   gwa\_pa0002-vorna.  
        if gwa\_pa0002-famst is not initial.  
          gwa\_output-marital  =   gwa\_pa0002-famst.  
        else.  
          gwa\_output-marital   =   ' '.  
        endif.  
  
\*       Gender  
        case gwa\_pa0002-gesch .  
          when  '1'.  
            gwa\_output-gender = 'M'.  
          when  '2'.  
            gwa\_output-gender = 'F'.  
          when others.  
            gwa\_output-gender = 'O'.  
        endcase.  
  
  
      endloop.  
  
  
\*   department  
      loop at gta\_pa0001 into gwa\_pa0001 where pernr = gwa\_pa0000-pernr and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda .  
  
        "        READ TABLE gta\_pa0001 INTO gwa\_pa0001  
        "        WITH  KEY pernr = gwa\_pa0002-pernr BINARY SEARCH.  
        data(lv\_orgeh)    = gwa\_pa0001-orgeh.  
        shift  lv\_orgeh left deleting leading '0'.  
        gwa\_output-depart = lv\_orgeh.  
  
        select single plstx into gwa\_output-occtitle  
          from t528t where sprsl = 'EN' and  
          plans =  gwa\_pa0001-plans     and  
          begda le gwa\_pa0000-endda     and  
          endda ge gwa\_pa0000-endda .  
      endloop.  
  
  
\*   Address  
      loop at gta\_pa0006 into gwa\_pa0006 where pernr = gwa\_pa0000-pernr and subty eq '1' and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda .  
        gwa\_output-address = gwa\_pa0006-stras.  
  
        concatenate gwa\_pa0006-ort01 gwa\_pa0006-state into gwa\_output-cityprov separated by ', '.  
        gwa\_output-pstlcode = gwa\_pa0006-pstlz.  
  
        data(lv\_telnr\_len)  = strlen( gwa\_pa0006-telnr ).  
\*      if strlen( gwa\_pa0006-telnr ) >= 10.  
\*\*        areacode  
\*        gwa\_output-areacode = gwa\_pa0006-telnr(3) .  
\*\*        phone  
\*        gwa\_output-phone = gwa\_pa0006-telnr+3(7).  
        if strlen( gwa\_pa0006-telnr ) >= 3.  
\*        areacode  
          gwa\_output-areacode = gwa\_pa0006-telnr(3) .  
\*        phone  
          gwa\_output-phone = gwa\_pa0006-telnr+3(lv\_telnr\_len).  
        elseif gwa\_pa0006-telnr is not initial.  
\*        areacode  
          gwa\_output-areacode = gwa\_pa0006-telnr(lv\_telnr\_len) .  
  
  
        endif.  
  
  
      endloop.  
  
\*   hoursperday and hoursperweek  
      loop at gta\_pa0007 into gwa\_pa0007 where pernr = gwa\_pa0000-pernr and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda .  
        gwa\_output-hoursperday =  gwa\_pa0007-arbst .  
        clear gv\_var.  
        gv\_var = gwa\_pa0007-wostd .  
        gwa\_output-hoursperweek = gv\_var(4) .  
      endloop.  
  
\*   salary, salarytyp and salarydate  
  
      loop at gta\_pa0008 into gwa\_pa0008 where pernr = gwa\_pa0000-pernr and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda .  
        clear: gv\_sal.  
        gv\_sal = gwa\_pa0008-ansal.  
        replace all occurrences of  '.' in  gv\_sal with ''.  
        condense   gv\_sal no-gaps.  
        gwa\_output-salary = gv\_sal.  
        gwa\_output-salarydate = gwa\_pa0008-begda .  
  
        if gwa\_pa0008-lga01 eq '1001'.  
          gwa\_output-salarytyp = 'w'.  
        elseif gwa\_pa0008-lga01 eq '1003'.  
          gwa\_output-salarytyp = 'A'.  
        else.  
          gwa\_output-salarytyp = 's'.  
        endif.  
      endloop.  
  
\*   union - Read IT0014  
      perform get\_union changing gwa\_output-union.  
  
\*   TD1 Exemption Amount  
      loop at gta\_pa0463 into gwa\_pa0463 where pernr = gwa\_pa0000-pernr and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda .  
  
        gwa\_output-td1expamnt = gwa\_pa0463-mntde.  
  
      endloop.  
  
\*   TD1 Exemption Code  
      perform get\_expcode changing gwa\_output-td1expcode.  
  
\*   termination date.  
      perform read\_date\_type  using gv\_begda gv\_endda '42'    changing gv\_last\_working\_date.  
      gwa\_output-termdate = gv\_last\_working\_date.  
  
      " Reformat all date fields.  
      perform reformat\_all\_date\_fields.  
  
      " Check field length and blank values  
      perform adjust\_field\_values.  
  
      append gwa\_output to gta\_output.  
      clear gwa\_output.  
    endloop.  
  endif.  
  
endform.  
form write\_csv.  
  
  
  
  concatenate p\_path p\_file into filename.  
  
  try.  
      open dataset filename for output in text mode encoding default.  
  
      loop at gta\_output into gwa\_output.  
        concatenate  
    gwa\_output-sin  
    gwa\_output-record\_type  
    gwa\_output-depart  
    gwa\_output-filler  
    gwa\_output-surname  
    gwa\_output-givename  
    gwa\_output-address  
    gwa\_output-cityprov  
    gwa\_output-pstlcode  
    gwa\_output-areacode  
    gwa\_output-phone  
    gwa\_output-dob  
    gwa\_output-marital  
    gwa\_output-gender  
    gwa\_output-occflag  
    gwa\_output-occtitle  
    gwa\_output-status  
    gwa\_output-empldate  
    gwa\_output-shiftstart  
    gwa\_output-shifteend  
    gwa\_output-ltdstart  
    gwa\_output-loastart  
    gwa\_output-loaend  
    gwa\_output-matstart  
    gwa\_output-matend  
    gwa\_output-empid  
    gwa\_output-healthnum  
    gwa\_output-yearsexp  
    gwa\_output-salary  
    gwa\_output-salarytyp  
    gwa\_output-salarydate  
    gwa\_output-language  
    gwa\_output-miscl  
    gwa\_output-union  
    gwa\_output-termdate  
    gwa\_output-td1expamnt  
    gwa\_output-td1expcode  
    gwa\_output-hoursperday  
    gwa\_output-hoursperweek  
    into gs\_text  
    respecting blanks. "separated by gc\_tab.  
\*          CONCATENATE gs\_text ';' INTO gs\_text.  
  
        append gs\_text to gt\_text.  
        transfer gs\_text to filename.  
  
      endloop.  
  
      close dataset filename.  
  
    catch cx\_root into data(lv\_exception) .  
\* Send error  
      data(lv\_local\_error)  = lv\_exception->get\_text( ) .  
      gv\_fatal\_error        = abap\_true.  
  
      concatenate gv\_error\_details lv\_local\_error into gv\_error\_details separated by gv\_lf.  
  
  endtry.  
endform.  
  
  
form alv\_report.  
  
\*Build field catalog  
  
  wa\_fieldcat-fieldname  = 'SIN'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'SIN'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'record\_type'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Record Typ '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'depart'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Department'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'filler'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Filler'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  
  
  wa\_fieldcat-fieldname  = 'surname'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Surname'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'givename'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Given Name'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'address'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Address'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'cityprov'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'City Prov'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'pstlcode'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Postal Code'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'areacode'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Area Code'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'phone'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Phone'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'dob'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Date of Birth'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'marital'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Marital Status'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'gender'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Gender'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'occflag'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Occuption Flag'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'occtitle'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Occuption Title'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'status'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Status'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'empldate'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Employment Date'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'shiftstart'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Shift Start'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'shiftend'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Shift End'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'ltdstart'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'LTD Start Date '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'loastart'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'LOA Start Date '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'loaend'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'LOA End Date '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'matedd'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Maternity Est Delivery Date  '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'matstart'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Materninty Start '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'mated'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Materninty End'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'empid'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Employee Number'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'healthnum'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Health Insurance Number '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'yearsexp'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Years Experience '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'salary'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Salary'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
\*  wa\_fieldcat-fieldname  = 'salary'.    " Fieldname in the data table  
\*  wa\_fieldcat-seltext\_m  = 'Salary'.   " Column description in the output  
\*  APPEND wa\_fieldcat TO it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'salarytyp'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Salary Type'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'salarydate'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Salary Date'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'language'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Language'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'miscl'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Miscellaneous '.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'union'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Union'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'termdate'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Termination Date'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'td1expamnt'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'TD1 Exemption Amount'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'td1expcode'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'TD1 Exemption Code'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'hoursperday'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Hours Worked Per Day'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  wa\_fieldcat-fieldname  = 'hoursperweek'.    " Fieldname in the data table  
  wa\_fieldcat-seltext\_m  = 'Hours Worked Per Week'.   " Column description in the output  
  append wa\_fieldcat to it\_fieldcat.  
  
  
  
\*Pass data and field catalog to ALV function module to display ALV list  
  call function 'REUSE\_ALV\_GRID\_DISPLAY'  
    exporting  
      it\_fieldcat   = it\_fieldcat  
    tables  
      t\_outtab      = gta\_output  
    exceptions  
      program\_error = 1  
      others        = 2.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  READ\_INFTY  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_gv\_begda  text  
\*      -->P\_gv\_endda  text  
\*      <--P\_gta\_pa0000  text  
\*----------------------------------------------------------------------\*  
form read\_infty  using    p\_begda  
                          p\_endda  
                          p\_infty  
                 changing p\_pnnnn type any table.  
  
  field-symbols: <lt\_pnnnn> type standard table.  
  
  assign p\_pnnnn to  <lt\_pnnnn>.  
  
  
  
  call function 'HR\_READ\_INFOTYPE'  
    exporting  
\*     TCLAS           = 'A'  
      pernr           = peras-pernr  
      infty           = p\_infty  
      begda           = p\_begda  
      endda           = p\_endda  
\*     SPRPS           = '\*'  
\*     BYPASS\_BUFFER   = ' '  
\*     LEGACY\_MODE     = ' '  
\*     IMPORTING  
\*     SUBRC           =  
    tables  
      infty\_tab       = <lt\_pnnnn>  
    exceptions  
      infty\_not\_found = 1  
      invalid\_input   = 2  
      others          = 3.  
  .  
  if sy-subrc <> 0.  
\*     Implement suitable error handling here  
  endif.  
  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  READ\_PERSONID  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_gv\_begda  text  
\*      -->P\_gv\_endda  text  
\*      -->P\_0052   text  
\*      <--P\_GV\_PERSONID  text  
\*----------------------------------------------------------------------\*  
form read\_personid  changing p\_personid.  
  
  call function 'GET\_PERSONID\_EXT\_BY\_PERNR'  
    exporting  
      pernr                  = peras-pernr  
    importing  
      personid\_ext           = p\_personid  
    exceptions  
      personid\_ext\_not\_found = 1  
      system\_error           = 2  
      others                 = 3.  
  
  if sy-subrc <> 0.  
\*       Implement suitable error handling here  
  
  endif.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  DATE\_SELECTION  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form date\_selection .  
  
  if pn-begda is not initial.  
    gv\_begda  = pn-begda.  
  else.  
    gv\_begda  = sy-datum.  
  endif.  
  
  if pn-endda is not initial.  
    gv\_endda  = pn-endda.  
  elseif gv\_begda gt sy-datum.  
    gv\_endda  = gv\_begda.  
  else.  
    gv\_endda  = sy-datum.  
  endif.  
  
  data:  lw\_begda\_range type line of lr\_date\_range.  
  
  lw\_begda\_range-sign     = 'I'.  
  lw\_begda\_range-option   = 'BT'.  
  lw\_begda\_range-low      = gv\_begda.  
  lw\_begda\_range-high     = gv\_endda.  
  
  append lw\_begda\_range to gt\_date\_range.  
  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  CLEANUP  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form cleanup .  
  
  clear: gta\_pa0000,  
         gwa\_pa0000,  
         gta\_pa0001,  
         gwa\_pa0001,  
         gta\_pa0002,  
         gwa\_pa0002,  
         gta\_pa0006,  
         gwa\_pa0006,  
         gta\_pa0007,  
         gwa\_pa0007,  
         gta\_pa0008,  
         gwa\_pa0008,  
         gta\_pa0014,  
         gwa\_pa0014,  
         gta\_pa0463,  
         gwa\_pa0463,  
         gta\_pa0464,  
         gwa\_pa0464,  
         gta\_pa0709,  
         gwa\_pa0709,  
         gta\_pa0712,  
         gwa\_output.  
  
  clear: gv\_personid,  
         gv\_entry\_date,  
         gv\_last\_working\_date,  
         gv\_sal,  
         gv\_pernr.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  CHECK\_ERRORS  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form check\_errors .  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  SEND\_ERROR\_REPORT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form send\_error\_report .  
  
  if gv\_fatal\_error eq abap\_true.  
  
    write: 'An exception was raised which couldn''t be handled. The file could not be generated. Error details below:' .  
    new-line.  
    write: gv\_error\_details.  
  
  endif.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  READ\_DATE\_TYPE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_GV\_BEGDA  text  
\*      -->P\_GV\_ENDDA  text  
\*      <--P\_GV\_ENTRY\_DATE  text  
\*----------------------------------------------------------------------\*  
form read\_date\_type  using    p\_begda  
                              p\_endda  
                              p\_date\_type  
                     changing p\_entry\_date.  
  
  
  data: lv\_msg\_handler type ref to  if\_hrpa\_message\_handler,  
        lv\_date        type p0041-dat01,  
        lv\_is\_ok       type boole\_d.  
  
  call function 'HR\_ECM\_GET\_DATETYP\_FROM\_IT0041'  
    exporting  
      pernr           = peras-pernr  
      keydt           = p\_endda  
      datar           = p\_date\_type  
      message\_handler = lv\_msg\_handler  
    importing  
      date            = lv\_date  
      is\_ok           = lv\_is\_ok.  
  if lv\_is\_ok eq abap\_true.  
    p\_entry\_date  = lv\_date.  
  else.  
    clear p\_entry\_date.  
  endif.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  GET\_YEAR\_FROM\_DATE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_GV\_ENTRY\_DATE  text  
\*      -->P\_SY\_DATUM  text  
\*      <--P\_GWA\_OUTPUT\_YEARSEXP  text  
\*----------------------------------------------------------------------\*  
form get\_year\_from\_date  using    p\_entry\_date  
                         changing p\_yearsexp.  
  
  data years\_between\_dates type string.  
  
  call function 'COMPUTE\_YEARS\_BETWEEN\_DATES'  
    exporting  
      first\_date                  = p\_entry\_date  
\*     MODIFY\_INTERVAL             = ' '  
      second\_date                 = sy-datum  
    importing  
      years\_between\_dates         = years\_between\_dates  
    exceptions  
      sequence\_of\_dates\_not\_valid = 1  
      others                      = 2.  
  if sy-subrc <> 0.  
\*     Implement suitable error handling here  
    p\_yearsexp  = '  '.  
  else.  
    condense years\_between\_dates no-gaps.  
    p\_yearsexp  = years\_between\_dates.  
  endif.  
  
  call function 'CONVERSION\_EXIT\_ALPHA\_INPUT'  
    exporting  
      input  = p\_yearsexp  
    importing  
      output = p\_yearsexp.  
  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  GET\_UNION  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      <--P\_GWA\_OUTPUT\_UNION  text  
\*----------------------------------------------------------------------\*  
form get\_union  changing p\_union.  
  
  
  data: lt\_lgart type  lr\_wagetype\_range,  
        lw\_lgart type line of lr\_wagetype\_range,  
        lv\_molga type t500l-molga,  
        lv\_text  type c length 25.  
  
  lw\_lgart-sign     = 'I'.  
  lw\_lgart-option   = 'BT'.  
  lw\_lgart-low      = '4000'.  
  lw\_lgart-high     = '4029'.  
  
  append lw\_lgart to lt\_lgart.  
  
  perform get\_molga changing lv\_molga.  
  
  data:  lw\_begda\_range type line of lr\_date\_range.  
  
  clear gt\_date\_range.  
  
  lw\_begda\_range-sign     = 'I'.  
  lw\_begda\_range-option   = 'BT'.  
  lw\_begda\_range-low      = gwa\_pa0000-begda.  
  lw\_begda\_range-high     = gwa\_pa0000-endda.  
  
  append lw\_begda\_range to gt\_date\_range.  
  
  
  loop at gta\_pa0014 into gwa\_pa0014 where pernr = gwa\_pa0000-pernr and begda in gt\_date\_range and lgart in lt\_lgart.  
  
    "T512T  
    select single lgtxt from t512t into lv\_text where sprsl eq 'E' and  lgart eq gwa\_pa0014-lgart and molga eq lv\_molga.  
    if sy-subrc eq 0.  
      exit.  
    endif.  
  
  endloop.  
  p\_union = lv\_text.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  GET\_MOLGA  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      <--P\_LV\_MOLGA  text  
\*----------------------------------------------------------------------\*  
form get\_molga  changing p\_molga.  
  
  call function 'RH\_PM\_GET\_MOLGA\_FROM\_PERNR'  
    exporting  
\*     PLVAR           = PLVAR  
      pernr           = peras-pernr  
      begda           = gv\_begda  
      endda           = gv\_endda  
    importing  
      molga           = p\_molga  
\*     TRFKZ           = TRFKZ  
    exceptions  
      nothing\_found   = 1  
      no\_active\_plvar = 2  
      others          = 3.  
  if sy-subrc <> 0.  
\*     Implement suitable error handling here  
  endif.  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  REFORMAT\_DATE\_FIELDS  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_GWA\_OUTPUT\_DOB  text  
\*      <--P\_GWA\_OUTPUT\_DOB  text  
\*----------------------------------------------------------------------\*  
form reformat\_date\_fields  using    p\_input\_date  
                           changing p\_output\_date.  
  
  data lv\_date type datum.  
  if p\_input\_date is not initial.  
    lv\_date = p\_input\_date.  
    call function 'DATE\_CHECK\_PLAUSIBILITY'  
      exporting  
        date                      = lv\_date  
      exceptions  
        plausibility\_check\_failed = 1  
        others                    = 2.  
    if sy-subrc <> 0.  
\*     Implement suitable error handling here  
    else.  
      concatenate lv\_date+6(2) lv\_date+4(2) lv\_date(4) into p\_output\_date.  
    endif.  
  
  
  
  endif.  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  REFORMAT\_ALL\_DATE\_FIELDS  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form reformat\_all\_date\_fields .  
  
\*   Date of Birth  
  perform reformat\_date\_fields using gwa\_output-dob changing gwa\_output-dob.  
  
\*   Employment Date  
  perform reformat\_date\_fields using gwa\_output-empldate changing gwa\_output-empldate.  
  
\*   LTD Start Date  
  perform reformat\_date\_fields using gwa\_output-ltdstart changing gwa\_output-ltdstart.  
  
\*   LOA Start Date  
  perform reformat\_date\_fields using gwa\_output-loastart changing gwa\_output-loastart.  
  
\*   LOA End Date  
  perform reformat\_date\_fields using gwa\_output-loaend changing gwa\_output-loaend.  
  
\*   Maternity EDD  
  perform reformat\_date\_fields using gwa\_output-matedd changing gwa\_output-matedd.  
  
\*   Maternity LOA Start Date  
  perform reformat\_date\_fields using gwa\_output-matstart changing gwa\_output-matstart.  
  
\*   Maternity LOA End Date  
  perform reformat\_date\_fields using gwa\_output-matend changing gwa\_output-matend.  
  
\*   Salary Date  
  perform reformat\_date\_fields using gwa\_output-salarydate changing gwa\_output-salarydate.  
  
\*   Termination Date  
  perform reformat\_date\_fields using gwa\_output-termdate changing gwa\_output-termdate.  
  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  GET\_EXPCODE  
\*&  Read indicator for WCB tax authority.  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form get\_expcode changing p\_exp\_code.  
  
  data: lv\_count      type  n,  
        lv\_count\_char type  c length 2,  
        lv\_field\_id   type  string,  
        lv\_exp\_code   type  pca\_exind.  
  
  field-symbols: <lw\_p0464>      type p0464,  
                 <fs\_0464\_value> type any.  
  
  loop at gta\_pa0464 assigning <lw\_p0464> where pernr = gwa\_pa0000-pernr and begda le gwa\_pa0000-endda and endda ge gwa\_pa0000-endda.  
    clear: lv\_count.  
    do 15 times.  
      lv\_count      = lv\_count + 1.  
      lv\_count\_char = lv\_count.  
  
      call function 'CONVERSION\_EXIT\_ALPHA\_INPUT'  
        exporting  
          input  = lv\_count\_char  
        importing  
          output = lv\_count\_char.  
  
  
      clear lv\_field\_id.  
      concatenate 'TAX' lv\_count\_char into lv\_field\_id.  
      assign component lv\_field\_id of structure <lw\_p0464> to <fs\_0464\_value>.  
      if <fs\_0464\_value> is assigned and <fs\_0464\_value> eq 'WCB'.  
  
        unassign <fs\_0464\_value>.  
        clear lv\_field\_id.  
        concatenate 'EXI' lv\_count\_char into lv\_field\_id.  
        assign component lv\_field\_id of structure <lw\_p0464> to <fs\_0464\_value>.  
        if <fs\_0464\_value> is assigned.  
          lv\_exp\_code = <fs\_0464\_value>.  
        endif.  
        exit.  
  
      endif.  
  
    enddo.  
  endloop.  
  p\_exp\_code = lv\_exp\_code.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  FILL\_CONSTANT\_VALUES  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form fill\_constant\_values .  
  
  gwa\_output-record\_type = '#' .  
  gwa\_output-filler = '    ' .  
  gwa\_output-matedd = '        '.  
  gwa\_output-shiftstart = '    '.  
  gwa\_output-shifteend = '    '.  
\*  gwa\_output-matend = '    '.  
  gwa\_output-healthnum  = '            '.  
  
  gwa\_output-miscl = '                         '.  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  SET\_SEL\_SCREEN\_PROPERTIES  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form set\_sel\_screen\_properties .  
  loop at screen.  
  
    if screen-group1 = 'RO'.  
      screen-input = 0.  
      modify screen. " Read only  
    endif.  
  
  endloop.  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  ADJUST\_FIELD\_VALUES  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form adjust\_field\_values .  
  data: lv\_table\_descr  type ref to cl\_abap\_tabledescr,  
        lv\_struct\_descr type ref to cl\_abap\_structdescr,  
        lv\_columns      type abap\_compdescr\_tab,  
        lv\_space(1)     value ' '.  
  
  
  field-symbols:  
    <lv\_column> like line of lv\_columns,  
    <lv\_field>  type any.  
  try.  
      lv\_table\_descr ?= cl\_abap\_typedescr=>describe\_by\_data( gta\_output ).  
    catch cx\_root .  
  endtry.  
  try .  
      lv\_struct\_descr ?= lv\_table\_descr->get\_table\_line\_type( ).  
      lv\_columns = lv\_struct\_descr->components.  
  
      loop at lv\_columns assigning <lv\_column>.  
        data(lv\_field\_length) = <lv\_column>-length / 2.  
        assign component <lv\_column>-name of structure gwa\_output to <lv\_field>.  
        if <lv\_field> is assigned.  
          if strlen( <lv\_field> ) < lv\_field\_length.  
            do ( lv\_field\_length - strlen( <lv\_field> ) ) times.  
              concatenate <lv\_field> lv\_space into <lv\_field> respecting blanks.  
  
            enddo.  
  
          endif.  
        endif.  
      endloop.  
    catch cx\_root.  
  endtry.  
  
  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*&      Form  CHECK\_MAIN\_PERNR  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
form check\_main\_pernr changing p\_is\_main\_pernr type boole\_d.  
  "data lv\_is\_main\_pernr type boole\_d value ''.  
  
  perform read\_infty  using gv\_begda gv\_endda '0712'  changing gta\_pa0712.  
  if gta\_pa0712 is initial. "Pernr is main if no IT0712 exists. This scenario should not occur in prod, i.e. all PERNRs have IT0712 created.  
    p\_is\_main\_pernr = 'X'.  
  else.  
    loop at gta\_pa0712 into gwa\_pa0712 where subty eq 'MAIN'.  
      if peras-pernr eq gwa\_pa0712-main\_pernr.  
        p\_is\_main\_pernr = 'X'.  
      endif.  
    endloop.  
  endif.  
endform.