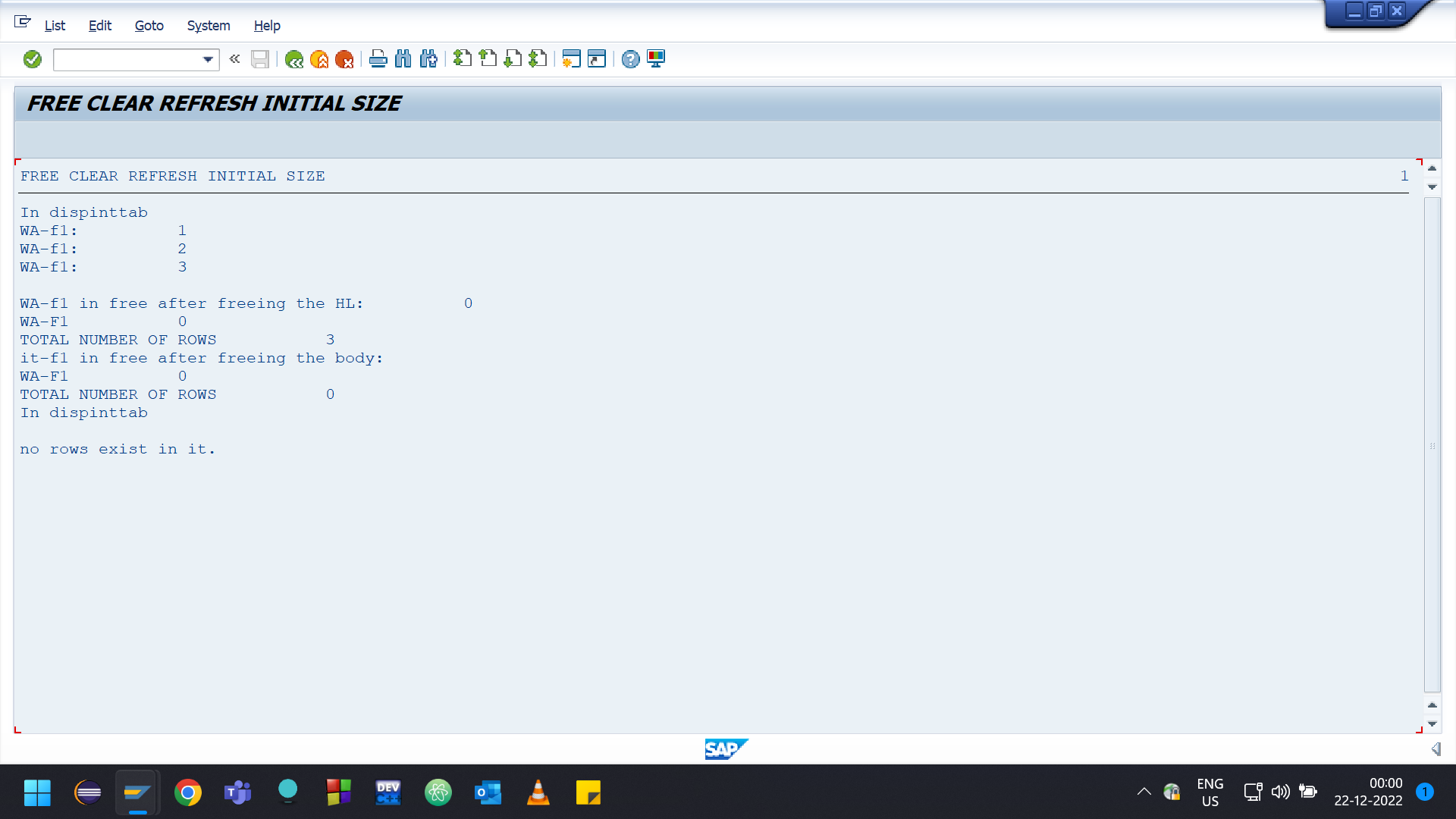
\*&---------------------------------------------------------------------\*  
\*& Report  ZYNY\_INTERNAL\_TABLE\_07  
\*&  
\*&---------------------------------------------------------------------\*  
\*&  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZYNY\_INTERNAL\_TABLE\_07.  
  
types: begin of it\_typ,  
        f1  TYPE i,  
       end of it\_typ.  
  
data : it type standard table of it\_typ  INITIAL SIZE  0.  
data wa type it\_typ.  
DATA NUMBER TYPE I.  
  
\*do  3 times.  
\*  it-f1 = sy-index.  
\*  append it.  
\*enddo.  
  
do  3 times.  
  CLEAR WA.  
  WA-f1 = sy-index.  
  append WA TO it.  
enddo.  
  
PERFORM dispinttab.  
  
\*PERFORM freeinttabWA. " IT DOES NOT DELETE WORK AREA, ONLY BODY  
\*PERFORM freeinttabBODY.  
\*PERFORM dispinttab. " BUT AT THE END IT WAS NOT TELLING SOME ERROR, LIKE INITIAL IS NOT WORKING  
\* ON EMPTY INTERNAL ISSUE, MAY BE BECAUSE OF INITIAL SISE STATEMENT  
  
\*PERFORM clearinttabWA. " IT CAN DELETE BOTH WORK AREA AND BODY  
\*PERFORM clearinttabBODY.  
\*PERFORM dispinttab. " INITIAL IS WORKING  
  
PERFORM refreshinttabWA. " REFRESH DOES NOT WORK ON WORK AREA  
PERFORM refreshinttabBODY.  
PERFORM dispinttab.  
  
if it is initial.  
  write:/ 'no rows exist in it.'.  
ELSE.  
  write:/ 'rows exist in it.'.  
ENDIF.  
  
  
FORM dispinttab.  
  write:/ 'In dispinttab'.  
  loop at it INTO WA.  
    write: / 'WA-f1:' , WA-f1.  
  endloop.  
  WRITE : /.  
ENDFORM.  
  
form freeinttabWA.  
    free WA.  
    write :/'WA-f1 in free after freeing the HL:' ,WA-f1.  
    WRITE : / 'WA-F1 ', WA-F1.  
    PERFORM GETROWNUMBERS.  
ENDFORM.  
  
form freeinttabBODY.  
    free it. " WITHOUT []  
    write :/'it-f1 in free after freeing the body:'.  
    READ TABLE IT INTO WA INDEX 1.  
    WRITE : / 'WA-F1 ', WA-F1.  
    PERFORM GETROWNUMBERS.  
  
\*    free it[]. " WITH []  
\*    write :/'it-f1 in free after freeing the body:'.  
\*    READ TABLE IT INTO WA INDEX 1.  
\*    WRITE : / 'WA-F1 ', WA-F1.  
\*    PERFORM GETROWNUMBERS.  
endform.  
  
FORM clearinttabWA.  
clear WA.  
write :/'WA-f1 in clear after clearing the HL:' ,WA-f1.  
    WRITE : / 'WA-F1 ', WA-F1.  
    PERFORM GETROWNUMBERS.  
ENDFORM.  
  
FORM clearinttabBODY.  
clear it.  
write :/'it-f1 in clear after clearing the body:'.  
    READ TABLE IT INTO WA INDEX 1.  
    WRITE : / 'WA-F1 ', WA-F1.  
    PERFORM GETROWNUMBERS.  
endform.  
  
FORM  refreshinttabWA.  
\*refresh WA.  
write :/'WA-f1 in after refreshing the HL:' ,WA-f1.  
    WRITE : / 'WA-F1 ', WA-F1.  
    PERFORM GETROWNUMBERS.  
ENDFORM.  
  
FORM  refreshinttabBODY.  
refresh it.  
write :/'it-f1 in after refreshing the body:'.  
    READ TABLE IT INTO WA INDEX 1.  
    WRITE : / 'WA-F1 ', WA-F1.  
    PERFORM GETROWNUMBERS.  
endform.  
  
FORM GETROWNUMBERS.  
  NUMBER = 0.  
  DESCRIBE TABLE IT LINES NUMBER.  
  WRITE : / 'TOTAL NUMBER OF ROWS ', NUMBER.  
ENDFORM.

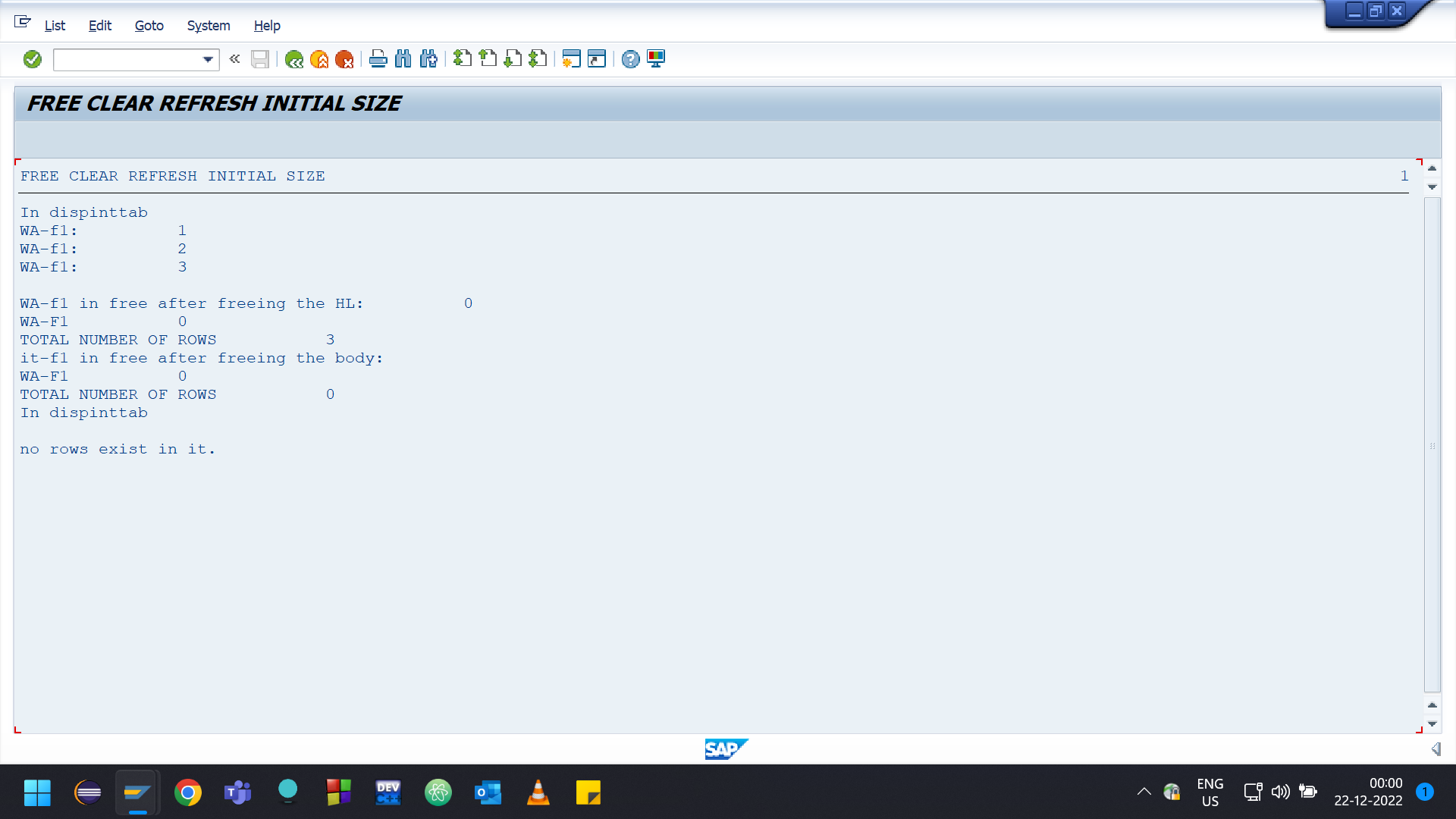
WITHOUT HEADER LINE INITIAL SIZE

FREE DEMO

IT WILL WORK WITH BOTH WITH [] OR WITHOUT [] TO FREE A INTERNAL TABLE

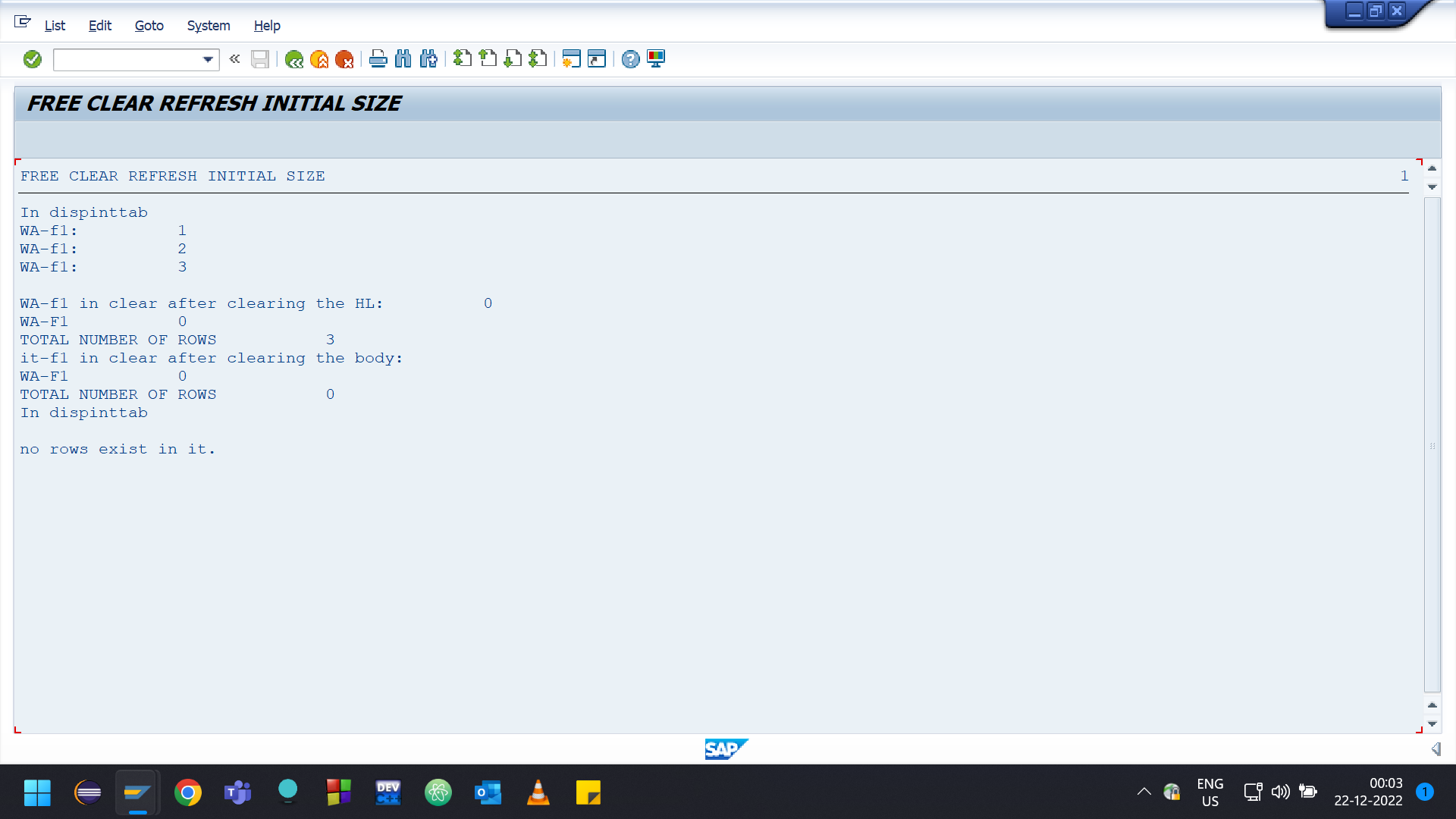


WITH []



CLEAR DEMO

IT WILL WORK WITH BOTH WITH [] OR WITHOUT [] TO FREE A INTERNAL TABLE



REFRESH DEMO

