\*&---------------------------------------------------------------------\*  
\*&  Include           ZFVOPFF05\_PROJETO\_CLI\_TC  
\*&---------------------------------------------------------------------\*  
  
\*&SPWIZARD: DECLARATION OF TABLECONTROL 'TC\_CLIENTE' ITSELF  
CONTROLS: TC\_CLIENTE TYPE TABLEVIEW USING SCREEN 9180.  
  
\*&SPWIZARD: LINES OF TABLECONTROL 'TC\_CLIENTE'  
DATA:     G\_TC\_CLIENTE\_LINES  LIKE SY-LOOPC.  
  
DATA:     OK\_CODE LIKE SY-UCOMM.  
  
\*&SPWIZARD: OUTPUT MODULE FOR TC 'TC\_CLIENTE'. DO NOT CHANGE THIS LINE!  
\*&SPWIZARD: UPDATE LINES FOR EQUIVALENT SCROLLBAR  
MODULE TC\_CLIENTE\_CHANGE\_TC\_ATTR OUTPUT.  
  DESCRIBE TABLE TI\_ZFVTPFF05\_CLIENT LINES TC\_CLIENTE-lines.  
ENDMODULE.  
  
\*&SPWIZARD: OUTPUT MODULE FOR TC 'TC\_CLIENTE'. DO NOT CHANGE THIS LINE!  
\*&SPWIZARD: GET LINES OF TABLECONTROL  
MODULE TC\_CLIENTE\_GET\_LINES OUTPUT.  
  G\_TC\_CLIENTE\_LINES = SY-LOOPC.  
ENDMODULE.  
  
\*&SPWIZARD: INPUT MODULE FOR TC 'TC\_CLIENTE'. DO NOT CHANGE THIS LINE!  
\*&SPWIZARD: PROCESS USER COMMAND  
MODULE TC\_CLIENTE\_USER\_COMMAND INPUT.  
  OK\_CODE = SY-UCOMM.  
  PERFORM USER\_OK\_TC USING    'TC\_CLIENTE'  
                              'TI\_ZFVTPFF05\_CLIENT'  
                              ' '  
                     CHANGING OK\_CODE.  
  SY-UCOMM = OK\_CODE.  
ENDMODULE.  
  
\*----------------------------------------------------------------------\*  
\*   INCLUDE TABLECONTROL\_FORMS                                         \*  
\*----------------------------------------------------------------------\*  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  USER\_OK\_TC                                               \*  
\*&---------------------------------------------------------------------\*  
 FORM USER\_OK\_TC USING    P\_TC\_NAME TYPE DYNFNAM  
                          P\_TABLE\_NAME  
                          P\_MARK\_NAME  
                 CHANGING P\_OK      LIKE SY-UCOMM.  
  
\*&SPWIZARD: BEGIN OF LOCAL DATA----------------------------------------\*  
   DATA: L\_OK              TYPE SY-UCOMM,  
         L\_OFFSET          TYPE I.  
\*&SPWIZARD: END OF LOCAL DATA------------------------------------------\*  
  
\*&SPWIZARD: Table control specific operations                          \*  
\*&SPWIZARD: evaluate TC name and operations                            \*  
   SEARCH P\_OK FOR P\_TC\_NAME.  
   IF SY-SUBRC <> 0.  
     EXIT.  
   ENDIF.  
   L\_OFFSET = STRLEN( P\_TC\_NAME ) + 1.  
   L\_OK = P\_OK+L\_OFFSET.  
\*&SPWIZARD: execute general and TC specific operations                 \*  
   CASE L\_OK.  
     WHEN 'INSR'.                      "insert row  
       PERFORM FCODE\_INSERT\_ROW USING    P\_TC\_NAME  
                                         P\_TABLE\_NAME.  
       CLEAR P\_OK.  
  
     WHEN 'DELE'.                      "delete row  
       PERFORM FCODE\_DELETE\_ROW USING    P\_TC\_NAME  
                                         P\_TABLE\_NAME  
                                         P\_MARK\_NAME.  
       CLEAR P\_OK.  
  
     WHEN 'P--' OR                     "top of list  
          'P-'  OR                     "previous page  
          'P+'  OR                     "next page  
          'P++'.                       "bottom of list  
       PERFORM COMPUTE\_SCROLLING\_IN\_TC USING P\_TC\_NAME  
                                             L\_OK.  
       CLEAR P\_OK.  
\*     WHEN 'L--'.                       "total left  
\*       PERFORM FCODE\_TOTAL\_LEFT USING P\_TC\_NAME.  
\*  
\*     WHEN 'L-'.                        "column left  
\*       PERFORM FCODE\_COLUMN\_LEFT USING P\_TC\_NAME.  
\*  
\*     WHEN 'R+'.                        "column right  
\*       PERFORM FCODE\_COLUMN\_RIGHT USING P\_TC\_NAME.  
\*  
\*     WHEN 'R++'.                       "total right  
\*       PERFORM FCODE\_TOTAL\_RIGHT USING P\_TC\_NAME.  
\*  
     WHEN 'MARK'.                      "mark all filled lines  
       PERFORM FCODE\_TC\_MARK\_LINES USING P\_TC\_NAME  
                                         P\_TABLE\_NAME  
                                         P\_MARK\_NAME   .  
       CLEAR P\_OK.  
  
     WHEN 'DMRK'.                      "demark all filled lines  
       PERFORM FCODE\_TC\_DEMARK\_LINES USING P\_TC\_NAME  
                                           P\_TABLE\_NAME  
                                           P\_MARK\_NAME .  
       CLEAR P\_OK.  
  
\*     WHEN 'SASCEND'   OR  
\*          'SDESCEND'.                  "sort column  
\*       PERFORM FCODE\_SORT\_TC USING P\_TC\_NAME  
\*                                   l\_ok.  
  
   ENDCASE.  
  
 ENDFORM.                              " USER\_OK\_TC  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  FCODE\_INSERT\_ROW                                         \*  
\*&---------------------------------------------------------------------\*  
 FORM fcode\_insert\_row  
               USING    P\_TC\_NAME           TYPE DYNFNAM  
                        P\_TABLE\_NAME             .  
  
\*&SPWIZARD: BEGIN OF LOCAL DATA----------------------------------------\*  
   DATA L\_LINES\_NAME       LIKE FELD-NAME.  
   DATA L\_SELLINE          LIKE SY-STEPL.  
   DATA L\_LASTLINE         TYPE I.  
   DATA L\_LINE             TYPE I.  
   DATA L\_TABLE\_NAME       LIKE FELD-NAME.  
   FIELD-SYMBOLS <TC>                 TYPE CXTAB\_CONTROL.  
   FIELD-SYMBOLS <TABLE>              TYPE STANDARD TABLE.  
   FIELD-SYMBOLS <LINES>              TYPE I.  
\*&SPWIZARD: END OF LOCAL DATA------------------------------------------\*  
  
   ASSIGN (P\_TC\_NAME) TO <TC>.  
  
\*&SPWIZARD: get the table, which belongs to the tc                     \*  
   CONCATENATE P\_TABLE\_NAME '[]' INTO L\_TABLE\_NAME. "table body  
   ASSIGN (L\_TABLE\_NAME) TO <TABLE>.                "not headerline  
  
\*&SPWIZARD: get looplines of TableControl                              \*  
   CONCATENATE 'G\_' P\_TC\_NAME '\_LINES' INTO L\_LINES\_NAME.  
   ASSIGN (L\_LINES\_NAME) TO <LINES>.  
  
\*&SPWIZARD: get current line                                           \*  
   GET CURSOR LINE L\_SELLINE.  
   IF SY-SUBRC <> 0.                   " append line to table  
     L\_SELLINE = <TC>-LINES + 1.  
\*&SPWIZARD: set top line                                               \*  
     IF L\_SELLINE > <LINES>.  
       <TC>-TOP\_LINE = L\_SELLINE - <LINES> + 1 .  
     ELSE.  
       <TC>-TOP\_LINE = 1.  
     ENDIF.  
   ELSE.                               " insert line into table  
     L\_SELLINE = <TC>-TOP\_LINE + L\_SELLINE - 1.  
     L\_LASTLINE = <TC>-TOP\_LINE + <LINES> - 1.  
   ENDIF.  
\*&SPWIZARD: set new cursor line                                        \*  
   L\_LINE = L\_SELLINE - <TC>-TOP\_LINE + 1.  
  
\*&SPWIZARD: insert initial line                                        \*  
   INSERT INITIAL LINE INTO <TABLE> INDEX L\_SELLINE.  
   <TC>-LINES = <TC>-LINES + 1.  
\*&SPWIZARD: set cursor                                                 \*  
   SET CURSOR LINE L\_LINE.  
  
 ENDFORM.                              " FCODE\_INSERT\_ROW  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  FCODE\_DELETE\_ROW                                         \*  
\*&---------------------------------------------------------------------\*  
 FORM fcode\_delete\_row  
               USING    P\_TC\_NAME           TYPE DYNFNAM  
                        P\_TABLE\_NAME  
                        P\_MARK\_NAME   .  
  
\*&SPWIZARD: BEGIN OF LOCAL DATA----------------------------------------\*  
   DATA L\_TABLE\_NAME       LIKE FELD-NAME.  
  
   FIELD-SYMBOLS <TC>         TYPE cxtab\_control.  
   FIELD-SYMBOLS <TABLE>      TYPE STANDARD TABLE.  
   FIELD-SYMBOLS <WA>.  
   FIELD-SYMBOLS <MARK\_FIELD>.  
\*&SPWIZARD: END OF LOCAL DATA------------------------------------------\*  
  
   ASSIGN (P\_TC\_NAME) TO <TC>.  
  
\*&SPWIZARD: get the table, which belongs to the tc                     \*  
   CONCATENATE P\_TABLE\_NAME '[]' INTO L\_TABLE\_NAME. "table body  
   ASSIGN (L\_TABLE\_NAME) TO <TABLE>.                "not headerline  
  
\*&SPWIZARD: delete marked lines                                        \*  
   DESCRIBE TABLE <TABLE> LINES <TC>-LINES.  
  
   LOOP AT <TABLE> ASSIGNING <WA>.  
  
\*&SPWIZARD: access to the component 'FLAG' of the table header         \*  
     ASSIGN COMPONENT P\_MARK\_NAME OF STRUCTURE <WA> TO <MARK\_FIELD>.  
  
     IF <MARK\_FIELD> = 'X'.  
       DELETE <TABLE> INDEX SYST-TABIX.  
       IF SY-SUBRC = 0.  
         <TC>-LINES = <TC>-LINES - 1.  
       ENDIF.  
     ENDIF.  
   ENDLOOP.  
  
 ENDFORM.                              " FCODE\_DELETE\_ROW  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  COMPUTE\_SCROLLING\_IN\_TC  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_TC\_NAME  name of tablecontrol  
\*      -->P\_OK       ok code  
\*----------------------------------------------------------------------\*  
 FORM COMPUTE\_SCROLLING\_IN\_TC USING    P\_TC\_NAME  
                                       P\_OK.  
\*&SPWIZARD: BEGIN OF LOCAL DATA----------------------------------------\*  
   DATA L\_TC\_NEW\_TOP\_LINE     TYPE I.  
   DATA L\_TC\_NAME             LIKE FELD-NAME.  
   DATA L\_TC\_LINES\_NAME       LIKE FELD-NAME.  
   DATA L\_TC\_FIELD\_NAME       LIKE FELD-NAME.  
  
   FIELD-SYMBOLS <TC>         TYPE cxtab\_control.  
   FIELD-SYMBOLS <LINES>      TYPE I.  
\*&SPWIZARD: END OF LOCAL DATA------------------------------------------\*  
  
   ASSIGN (P\_TC\_NAME) TO <TC>.  
\*&SPWIZARD: get looplines of TableControl                              \*  
   CONCATENATE 'G\_' P\_TC\_NAME '\_LINES' INTO L\_TC\_LINES\_NAME.  
   ASSIGN (L\_TC\_LINES\_NAME) TO <LINES>.  
  
  
\*&SPWIZARD: is no line filled?                                         \*  
   IF <TC>-LINES = 0.  
\*&SPWIZARD: yes, ...                                                   \*  
     L\_TC\_NEW\_TOP\_LINE = 1.  
   ELSE.  
\*&SPWIZARD: no, ...                                                    \*  
     CALL FUNCTION 'SCROLLING\_IN\_TABLE'  
          EXPORTING  
               ENTRY\_ACT             = <TC>-TOP\_LINE  
               ENTRY\_FROM            = 1  
               ENTRY\_TO              = <TC>-LINES  
               LAST\_PAGE\_FULL        = 'X'  
               LOOPS                 = <LINES>  
               OK\_CODE               = P\_OK  
               OVERLAPPING           = 'X'  
          IMPORTING  
               ENTRY\_NEW             = L\_TC\_NEW\_TOP\_LINE  
          EXCEPTIONS  
\*              NO\_ENTRY\_OR\_PAGE\_ACT  = 01  
\*              NO\_ENTRY\_TO           = 02  
\*              NO\_OK\_CODE\_OR\_PAGE\_GO = 03  
               OTHERS                = 0.  
   ENDIF.  
  
\*&SPWIZARD: get actual tc and column                                   \*  
   GET CURSOR FIELD L\_TC\_FIELD\_NAME  
              AREA  L\_TC\_NAME.  
  
   IF SYST-SUBRC = 0.  
     IF L\_TC\_NAME = P\_TC\_NAME.  
\*&SPWIZARD: et actual column                                           \*  
       SET CURSOR FIELD L\_TC\_FIELD\_NAME LINE 1.  
     ENDIF.  
   ENDIF.  
  
\*&SPWIZARD: set the new top line                                       \*  
   <TC>-TOP\_LINE = L\_TC\_NEW\_TOP\_LINE.  
  
  
 ENDFORM.                              " COMPUTE\_SCROLLING\_IN\_TC  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  FCODE\_TC\_MARK\_LINES  
\*&---------------------------------------------------------------------\*  
\*       marks all TableControl lines  
\*----------------------------------------------------------------------\*  
\*      -->P\_TC\_NAME  name of tablecontrol  
\*----------------------------------------------------------------------\*  
FORM FCODE\_TC\_MARK\_LINES USING P\_TC\_NAME  
                               P\_TABLE\_NAME  
                               P\_MARK\_NAME.  
\*&SPWIZARD: EGIN OF LOCAL DATA-----------------------------------------\*  
  DATA L\_TABLE\_NAME       LIKE FELD-NAME.  
  
  FIELD-SYMBOLS <TC>         TYPE cxtab\_control.  
  FIELD-SYMBOLS <TABLE>      TYPE STANDARD TABLE.  
  FIELD-SYMBOLS <WA>.  
  FIELD-SYMBOLS <MARK\_FIELD>.  
\*&SPWIZARD: END OF LOCAL DATA------------------------------------------\*  
  
  ASSIGN (P\_TC\_NAME) TO <TC>.  
  
\*&SPWIZARD: get the table, which belongs to the tc                     \*  
   CONCATENATE P\_TABLE\_NAME '[]' INTO L\_TABLE\_NAME. "table body  
   ASSIGN (L\_TABLE\_NAME) TO <TABLE>.                "not headerline  
  
\*&SPWIZARD: mark all filled lines                                      \*  
  LOOP AT <TABLE> ASSIGNING <WA>.  
  
\*&SPWIZARD: access to the component 'FLAG' of the table header         \*  
     ASSIGN COMPONENT P\_MARK\_NAME OF STRUCTURE <WA> TO <MARK\_FIELD>.  
  
     <MARK\_FIELD> = 'X'.  
  ENDLOOP.  
ENDFORM.                                          "fcode\_tc\_mark\_lines  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  FCODE\_TC\_DEMARK\_LINES  
\*&---------------------------------------------------------------------\*  
\*       demarks all TableControl lines  
\*----------------------------------------------------------------------\*  
\*      -->P\_TC\_NAME  name of tablecontrol  
\*----------------------------------------------------------------------\*  
FORM FCODE\_TC\_DEMARK\_LINES USING P\_TC\_NAME  
                                 P\_TABLE\_NAME  
                                 P\_MARK\_NAME .  
\*&SPWIZARD: BEGIN OF LOCAL DATA----------------------------------------\*  
  DATA L\_TABLE\_NAME       LIKE FELD-NAME.  
  
  FIELD-SYMBOLS <TC>         TYPE cxtab\_control.  
  FIELD-SYMBOLS <TABLE>      TYPE STANDARD TABLE.  
  FIELD-SYMBOLS <WA>.  
  FIELD-SYMBOLS <MARK\_FIELD>.  
\*&SPWIZARD: END OF LOCAL DATA------------------------------------------\*  
  
  ASSIGN (P\_TC\_NAME) TO <TC>.  
  
\*&SPWIZARD: get the table, which belongs to the tc                     \*  
   CONCATENATE P\_TABLE\_NAME '[]' INTO L\_TABLE\_NAME. "table body  
   ASSIGN (L\_TABLE\_NAME) TO <TABLE>.                "not headerline  
  
\*&SPWIZARD: demark all filled lines                                    \*  
  LOOP AT <TABLE> ASSIGNING <WA>.  
  
\*&SPWIZARD: access to the component 'FLAG' of the table header         \*  
     ASSIGN COMPONENT P\_MARK\_NAME OF STRUCTURE <WA> TO <MARK\_FIELD>.  
  
     <MARK\_FIELD> = SPACE.  
  ENDLOOP.  
ENDFORM.