\*&---------------------------------------------------------------------\*  
\*& Report ZREP\_EINV\_EWAY\_REPORT  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT zrep\_einv\_eway\_report.  
  
\* Type Pool declartion  
TYPE-POOLS: icon.  
  
  
\* Includes  
INCLUDE zrep\_einv\_eway\_report\_top.  
INCLUDE zrep\_einv\_eway\_report\_class.  
INCLUDE zrep\_einv\_eway\_report\_sel.  
INCLUDE zrep\_einv\_eway\_report\_forms.  
  
\* Initialization  
INITIALIZATION.  
  
\* Fetch User configuration based on selection  
  PERFORM fetch\_user\_config.  
  
\* Fetch all domain Values  
  PERFORM fetch\_domain\_values.  
  
  
AT SELECTION-SCREEN OUTPUT.  
  
\* Selection screen comments  
  WRITE icon\_message\_information AS ICON TO icon.  
  WRITE icon\_message\_information AS ICON TO icon1.  
  WRITE icon\_message\_information AS ICON TO icon2.  
  WRITE icon\_message\_information AS ICON TO icon3.  
  PERFORM set\_module\_value.  
  
  
\* Start of selection  
START-OF-SELECTION.  
  
\* Get all required data  
  PERFORM get\_data.  
  
\* If data available to dosplay call screen  
  IF gt\_final IS NOT INITIAL.  
    CALL SCREEN 9000.  
  ELSE.  
  
\* Message  
    MESSAGE 'No data for given selection'(m01) TYPE 'I'.  
    LEAVE LIST-PROCESSING.  
  
  ENDIF.  
  
  
\* Includes  
  INCLUDE zrep\_einv\_eway\_report\_statuo01.  
  
  INCLUDE zrep\_einv\_eway\_report\_user\_i01.

INCLUDE zrep\_einv\_eway\_report\_top.

\*&---------------------------------------------------------------------\*  
\*&  Include           ZREP\_EINV\_EWAY\_REPORT\_TOP  
\*&---------------------------------------------------------------------\*  
TYPE-POOLS: vrm.  
  
\* Type declarations  
  
TYPES: BEGIN OF ty\_zst\_einv\_api\_struct,  
         transporter\_id                 TYPE char20,  
         transporter\_name               TYPE char50,  
         transportation\_mode            TYPE zde\_t\_mode,  
         transportation\_distance        TYPE char10,  
         transporter\_document\_number    TYPE char15,  
         transporter\_document\_date      TYPE char10,  
         vehicle\_number                 TYPE char12,  
         vehicle\_type                   TYPE zde\_v\_type,  
         extend\_validity\_reason         TYPE zde\_ext\_reason,  
         extend\_remarks                 TYPE char100,  
         remaining\_distance             TYPE char10,  
         from\_pincode                   TYPE char10,  
         consignment\_status             TYPE zde\_consign\_sts,  
         transit\_type                   TYPE zde\_transit\_type,  
         address\_line1                  TYPE char50,  
         address\_line2                  TYPE char50,  
         address\_line3                  TYPE char50,  
         reason\_code\_for\_vehicle\_updati TYPE zde\_veh\_reason,  
         reason\_for\_vehicle\_updation    TYPE char50,  
         zport\_code                     TYPE char6,  
       END OF ty\_zst\_einv\_api\_struct.  
  
\* Type declarations  
TYPES : BEGIN OF ty\_wb2\_v\_vbrk\_vbrp2,  
          vbeln   TYPE vbeln\_vf,  
          vbeln\_i TYPE vbeln\_vf,  
          posnr\_i TYPE posnr\_vf,  
          fkart   TYPE fkart,  
          fktyp   TYPE fktyp,  
          vbtyp   TYPE fktyp,  
          waerk   TYPE waerk,  
          vkorg   TYPE  vkorg,  
          vtweg   TYPE  vtweg,  
          knumv   TYPE knumv,  
          fkdat   TYPE fkdat,  
          gjahr   TYPE gjahr,  
          land1   TYPE lland,  
          bukrs   TYPE bukrs,  
          netwr   TYPE netwr,  
          ernam   TYPE ernam,  
          kunrg   TYPE kunrg,  
          kunag   TYPE kunag,  
          xblnr   TYPE xblnr\_v1,  
          zuonr   TYPE ordnr\_v,  
          fksto   TYPE fksto,  
          bupla   TYPE bupla,  
          fkimg\_i TYPE fkimg,  
          meins\_i TYPE meins,  
          kursk\_i TYPE kursk,  
          netwr\_i TYPE netwr\_fp,  
          vgbel\_i TYPE vgbel,  
          matnr\_i TYPE matnr,  
          arktx\_i TYPE arktx,  
          charg\_i TYPE charg\_d,  
          pstyv\_i TYPE pstyv,  
          werks\_i TYPE werks\_d,  
          prctr\_i TYPE prctr,  
          mwsbp\_i TYPE mwsbp,  
        END OF ty\_wb2\_v\_vbrk\_vbrp2,  
        BEGIN OF ty\_wb2\_v\_likp\_lips2,  
          vbeln   TYPE  vbeln\_vl,  
          vbeln\_i  TYPE vbeln\_vl,  
          posnr\_i  TYPE posnr\_vl,  
          ernam   TYPE ernam,  
          vkorg    TYPE vkorg,  
          lfart    TYPE lfart,  
          lfdat    TYPE lfdat\_v,  
          vbtyp    TYPE vbtyp,  
          kunnr   TYPE kunwe,  
          kunag    TYPE kunag,  
          fkdat    TYPE fkdat,  
          knumv    TYPE knumv,  
          waerk    TYPE waerk,  
          lifnr    TYPE elifn,  
          bldat   TYPE   bldat,  
          netwr   TYPE  netwr\_ak,  
          werks    TYPE empfw,  
          xblnr    TYPE xblnr\_likp,  
          matnr\_i  TYPE matnr,  
          matkl\_i TYPE matkl,  
          werks\_i  TYPE werks\_d,  
          lgort\_i  TYPE lgort\_d,  
          charg\_i  TYPE charg\_d,  
          lfimg\_i  TYPE lfimg,  
          meins\_i  TYPE meins,  
          vrkme\_i  TYPE vrkme,  
          ntgew\_i  TYPE ntgew\_15,  
          brgew\_i  TYPE brgew\_15,  
          arktx\_i TYPE arktx,  
          vbelv\_i  TYPE vbelv,  
          posnv\_i  TYPE posnv,  
          vgbel\_i  TYPE vgbel,  
          vgpos\_i  TYPE vgpos,  
          gsber\_i  TYPE gsber,  
          prctr\_i  TYPE prctr,  
          netpr\_i  TYPE netpr,  
          netwr\_i TYPE netwr,  
          lfgja\_i TYPE lfgja,  
        END OF ty\_wb2\_v\_likp\_lips2,  
        BEGIN OF ty\_konv,  
          knumv  TYPE knumv,  
          kposn  TYPE kposn,  
          kschl TYPE kscha,  
          kbetr TYPE kbetr,  
          kwert TYPE kwert,  
        END OF ty\_konv,  
        BEGIN OF ty\_ewaybill,  
          bukrs     TYPE  bukrs,  
          doctyp    TYPE j\_1ig\_doctyp,  
          docno      TYPE j\_1ig\_docno,  
          gjahr      TYPE gjahr,  
          ebillno    TYPE j\_1ig\_ebillno,  
          egen\_dat  TYPE  j\_1ig\_egendat,  
          egen\_time  TYPE j\_1ig\_egentime,  
          vdfmdate  TYPE  j\_1ig\_vdfmdate,  
          vdtodate  TYPE  j\_1ig\_vdtodate,  
          vdtotime  TYPE  j\_1ig\_vdtotime,  
          status    TYPE j\_1ig\_stat,  
          ernam      TYPE ernam,  
          erdat     TYPE  erdat,  
          aenam      TYPE aenam,  
          aedat      TYPE aedat,  
        END OF ty\_ewaybill,  
        BEGIN OF ty\_marc,  
          matnr TYPE matnr,  
          werks TYPE werks\_d,  
          steuc TYPE steuc,  
        END OF ty\_marc,  
        BEGIN OF ty\_vbfa,  
          vbelv  TYPE vbeln\_von,  
          posnv  TYPE posnr\_von,  
          vbeln  TYPE vbeln\_nach,  
        END OF ty\_vbfa,  
        BEGIN OF ty\_wb2\_v\_vbak\_vbap2,  
          vbeln   TYPE vbeln\_va,  
          vbeln\_i TYPE vbeln\_va,  
          posnr\_i TYPE posnr\_va,  
          audat   TYPE audat,  
          bstnk   TYPE bstnk,  
          bstdk   TYPE bstdk,  
          ihrez   TYPE ihrez,  
          bname   TYPE bname\_v,  
        END OF ty\_wb2\_v\_vbak\_vbap2,  
        BEGIN OF ty\_gstin,  
          bukrs   TYPE bukrs,  
          branch TYPE j\_1bbranc\_,  
          gstin   TYPE j\_1igstcd3,  
        END OF ty\_gstin.  
  
TYPES: BEGIN OF ty\_kna1,  
         kunnr TYPE kunnr,  
         land1 TYPE land1\_gp,  
         name1 TYPE name1\_gp,  
         name2 TYPE name2\_gp,  
         ort01 TYPE ort01\_gp,  
         pstlz TYPE pstlz,  
         regio TYPE regio,  
         stras TYPE stras\_gp,  
         telf1 TYPE telf1,  
         adrnr TYPE  adrnr,  
         stcd3 TYPE stcd3,  
       END OF ty\_kna1.  
  
TYPES:BEGIN OF ty\_t005u,  
        spras  TYPE spras,  
        land1  TYPE land1,  
        bland  TYPE regio,  
        bezei  TYPE char100,  
      END OF ty\_t005u.  
  
  
TYPES:BEGIN OF ty\_adrc,  
        addrnumber TYPE ad\_addrnum,  
        tel\_number TYPE   ad\_tlnmbr1,  
        str\_suppl1 TYPE  adrc-str\_suppl1,  
        str\_suppl2 TYPE  adrc-str\_suppl2,  
      END OF ty\_adrc,  
      BEGIN OF   ty\_vbpa,  
        vbeln  TYPE vbeln,  
        posnr  TYPE posnr,  
        parvw  TYPE parvw,  
        kunnr  TYPE kunnr,  
        lifnr  TYPE lifnr,  
        adrnr  TYPE adrnr,  
      END OF ty\_vbpa.  
  
\*Dispatch and Ship to Party  
TYPES:BEGIN OF ty\_likp,  
        vbeln TYPE  vbeln\_vl,  
        kunnr  TYPE kunwe,  
      END OF ty\_likp.  
  
TYPES : BEGIN OF ty\_adr6,  
          addrnumber TYPE   ad\_addrnum,  
          smtp\_addr  TYPE ad\_smtpadr,  
        END OF ty\_adr6.  
  
TYPES:BEGIN OF ty\_doctyp,  
        bukrs      TYPE bukrs,  
        zmodule    TYPE zde\_zmodule,  
        fkart      TYPE zde\_docty,  
        vtext     TYPE   bezei40,  
        sup\_type  TYPE  zde\_sup\_type,  
        sub\_type  TYPE  zde\_sub\_type,  
        edoc\_type TYPE  zde\_edoc\_type,  
        einv      TYPE  zde\_einv,  
        eway      TYPE  zde\_eway,  
        accpost    TYPE zde\_accpost,  
        senderr    TYPE zde\_senderr,  
        instgen    TYPE zde\_instgen,  
        batch      TYPE zde\_batch,  
      END OF ty\_doctyp,  
      BEGIN OF ty\_user\_config,  
        bukrs   TYPE  bukrs,  
        zmodule  TYPE zde\_zmodule,  
        fkart    TYPE zde\_docty,  
        werks    TYPE werks\_d,  
        prctr    TYPE prctr,  
        gen\_apr  TYPE syuname,  
        blk\_usr  TYPE zde\_block,  
      END OF ty\_user\_config,  
      BEGIN OF ty\_api,  
        apiid   TYPE zde\_apiid,  
        apiprov TYPE zde\_apiprov,  
      END OF ty\_api,  
      BEGIN OF ty\_doc\_uom,  
        edoc\_uom TYPE zde\_edoc\_uom,  
        meins     TYPE meins,  
      END OF ty\_doc\_uom,  
      BEGIN OF ty\_export,  
        zport\_code     TYPE char6,  
        zport           TYPE char50,  
        land1           TYPE land1,  
        zport\_state     TYPE regio,  
        zport\_address1 TYPE  char100,  
        zport\_address2 TYPE  char100,  
        zport\_place     TYPE char30,  
        zport\_pincode   TYPE pstlz,  
      END OF ty\_export.  
  
TYPES: BEGIN OF ty\_final,  
         vbeln                TYPE vbrk-vbeln,  
         gjahr                TYPE vbrk-gjahr,  
         fkart                TYPE vbrk-fkart,  
         fktyp                TYPE vbrk-fktyp,  
         vbtyp                TYPE vbrk-vbtyp,  
         vkorg                TYPE vbrk-vkorg,  
         vtweg                TYPE vbrk-vtweg,  
         fkdat(10)            TYPE c,  
         kunrg                TYPE vbrk-kunrg,  
         check\_box(1) ,  
         ack\_no               TYPE  j\_1ig\_invrefnum-ack\_no,  
         ack\_dt               TYPE  j\_1ig\_invrefnum-ack\_date,  
         qr\_code              TYPE string,  
         irn                  TYPE string,  
         sign\_inv             TYPE string,  
         accs\_token           TYPE string,  
         canc\_dt              TYPE char10,  
         status               TYPE char10,  
         t\_id                 TYPE char20,  
         t\_name               TYPE char50,  
         t\_doc\_no             TYPE char15,  
         t\_date               TYPE char10,  
         t\_mode               TYPE zteway\_transport-t\_mode,  
         t\_distance           TYPE char10,  
         t\_r\_distance         TYPE char10,  
         t\_ext\_valid\_reason   TYPE zteway\_transport-t\_ext\_valid\_reason,  
         t\_ext\_valid\_remarks  TYPE char100,  
         t\_consignment\_status TYPE zteway\_transport-t\_consignment\_status,  
         t\_from\_pin           TYPE char10,  
         t\_transit\_type       TYPE zteway\_transport-t\_transit\_type,  
         t\_address1           TYPE char50,  
         t\_address2           TYPE char50,  
         t\_address3           TYPE char50,  
         v\_number             TYPE char12,  
         v\_type               TYPE zteway\_transport-v\_type,  
         eway\_print           TYPE string,  
         eway\_error           TYPE string,  
         v\_reason\_code        TYPE zteway\_transport-v\_reason\_code,  
         v\_reason             TYPE char50,  
         eway\_num             TYPE j\_1ig\_ebillno,  
         eway\_date            TYPE char10,  
         eway\_v\_from          TYPE char10,  
         eway\_v\_to            TYPE char10,  
         eway\_v\_time          TYPE char8,  
         eway\_status          TYPE char10,  
         eway\_canc\_dt         TYPE char10,  
         c\_reason\_code        TYPE zteway\_transport-c\_reason\_code,  
         c\_reason             TYPE char120,  
         werks                TYPE werks\_d,  
         prctr                TYPE prctr,  
         doc\_ernam            TYPE ernam,  
         ernam                TYPE ernam,  
         erdat                TYPE char10,  
         erzet                TYPE char8,  
         eway\_ernam           TYPE ernam,  
         eway\_erdat           TYPE char10,  
         eway\_erzet           TYPE char8,  
         bupla                TYPE bupla,  
         bukrs                TYPE bukrs,  
         sup\_gstin            TYPE j\_1igstcd3,  
         invval               TYPE dmbtr,  
         taxamt               TYPE dmbtr,  
         tax                  TYPE dmbtr,  
         odnno                TYPE xblnr\_v1,  
         kunnr                TYPE kunnr,  
         name1                TYPE name1\_gp,  
         ship\_to              TYPE kunnr,  
         ship\_to\_name         TYPE name1\_gp,  
         trans\_code           TYPE kunnr,  
         trans\_gstin          TYPE  j\_1igstcd3,  
         igst\_amt             TYPE dmbtr,  
         sgst\_amt             TYPE dmbtr,  
         cgst\_amt             TYPE dmbtr,  
         cess\_amt             TYPE dmbtr,  
         tcs\_amt             TYPE dmbtr,  
         icon(4)              TYPE c,  
         eway\_icon(4)         TYPE c,  
         fkdat\_db             TYPE vbrk-fkdat,  
         zport\_code           TYPE char6,  
         einv\_error           TYPE string,  
         e\_reason\_code        TYPE zteinv\_details-e\_reason\_code,  
         e\_reason             TYPE char120,  
         mod                  TYPE zteinv\_doctyp-zmodule,  
         einv                 TYPE xfeld,  
         eway                 TYPE xfeld,  
         ship\_gstin           TYPE char20,  
       END OF ty\_final,  
       BEGIN OF ty\_t001w1,  
         werks      TYPE t001w-werks,  
         name1      TYPE t001w-name1,  
         name2      TYPE t001w-name2,  
         stras      TYPE t001w-stras,  
         pstlz      TYPE t001w-pstlz,  
         ort01      TYPE t001w-ort01,  
         land1      TYPE t001w-land1,  
         regio      TYPE t001w-regio,  
         counc      TYPE t001w-counc,  
         cityc      TYPE t001w-cityc,  
         adrnr      TYPE t001w-adrnr,  
         spras      TYPE t001w-spras,  
         zone1      TYPE t001w-zone1,  
         j\_1bbranch TYPE t001w-j\_1bbranch,  
       END OF ty\_t001w1,  
       BEGIN OF ty\_j\_1bbranch,  
         bukrs  TYPE j\_1bbranch-bukrs,  
         branch TYPE j\_1bbranch-branch,  
         name   TYPE j\_1bbranch-name,  
         adrnr  TYPE j\_1bbranch-adrnr,  
         gstin  TYPE j\_1bbranch-gstin,  
       END OF ty\_j\_1bbranch,  
       BEGIN OF ty\_lfa1,  
         lifnr TYPE lfa1-lifnr,  
         land1 TYPE lfa1-land1,  
         name1 TYPE lfa1-name1,  
         name2 TYPE lfa1-name2,  
         name3 TYPE lfa1-name3,  
         name4 TYPE lfa1-name4,  
         ort01 TYPE lfa1-ort01,  
         pstlz TYPE lfa1-pstlz,  
         regio TYPE lfa1-regio,  
         stras TYPE lfa1-stras,  
         adrnr TYPE lfa1-adrnr,  
         stcd3 TYPE lfa1-stcd3,  
       END OF ty\_lfa1,  
       BEGIN OF ty\_bkpf1,  
         bukrs TYPE bkpf-bukrs,  
         belnr TYPE bkpf-belnr,  
         gjahr TYPE bkpf-gjahr,  
         blart TYPE bkpf-blart,  
         bldat TYPE bkpf-bldat,  
         budat TYPE bkpf-budat,  
         tcode TYPE bkpf-tcode,  
         xblnr TYPE bkpf-xblnr,  
         bktxt TYPE bkpf-bktxt,  
         awtyp TYPE bkpf-awtyp,  
         awkey TYPE bkpf-awkey,  
       END OF ty\_bkpf1,  
       BEGIN OF ty\_bseg1,  
         bukrs TYPE bseg-bukrs,  
         belnr TYPE bseg-belnr,  
         gjahr TYPE bseg-gjahr,  
         buzei TYPE bseg-buzei,  
         buzid TYPE bseg-buzid,  
         bschl TYPE bseg-bschl,  
         koart TYPE bseg-koart,  
         shkzg TYPE bseg-shkzg,  
         dmbtr TYPE bseg-dmbtr,  
         txgrp TYPE bseg-txgrp,  
         ktosl TYPE bseg-ktosl,  
         ebeln TYPE bseg-ebeln,  
         ebelp TYPE bseg-ebelp,  
         taxps TYPE bseg-taxps,  
       END OF ty\_bseg1,  
       BEGIN OF ty\_tvarvc,  
         name  TYPE rvari\_vnam,  
         type TYPE rsscr\_kind,  
         numb TYPE tvarv\_numb,  
         sign TYPE tvarv\_sign,  
         opti TYPE tvarv\_opti,  
         low  TYPE rvari\_val\_255,  
         high TYPE rvari\_val\_255,  
       END OF ty\_tvarvc.  
\*  FI data delcartion  
TYPES: BEGIN OF ty\_bkpf,  
         bukrs TYPE bkpf-bukrs,  
         belnr TYPE bkpf-belnr,  
         gjahr TYPE bkpf-gjahr,  
         blart TYPE bkpf-blart,  
         bldat TYPE bkpf-bldat,  
         budat TYPE bkpf-budat,  
         tcode TYPE bkpf-tcode,  
         usnam TYPE bkpf-tcode,  
         xblnr TYPE bkpf-xblnr,  
         bktxt TYPE bkpf-bktxt,  
         awtyp TYPE bkpf-awtyp,  
         awkey TYPE bkpf-awkey,  
       END OF ty\_bkpf,  
       BEGIN OF ty\_bseg,  
         bukrs    TYPE bseg-bukrs,  
         belnr    TYPE bseg-belnr,  
         gjahr    TYPE bseg-gjahr,  
         buzei    TYPE bseg-buzei,  
         buzid    TYPE bseg-buzid,  
         bschl    TYPE bseg-bschl,  
         koart    TYPE bseg-koart,  
         shkzg    TYPE bseg-shkzg,  
         dmbtr    TYPE bseg-dmbtr,  
         ktosl    TYPE bseg-ktosl,  
         sgtxt    TYPE bseg-sgtxt,  
         xauto    TYPE bseg-xauto,  
         kunnr    TYPE bseg-kunnr,  
         lifnr    TYPE bseg-lifnr,  
         matnr    TYPE bseg-matnr,  
         werks    TYPE bseg-werks,  
         menge    TYPE bseg-menge,  
         meins    TYPE bseg-meins,  
         ebeln    TYPE bseg-ebeln,  
         ebelp    TYPE bseg-ebelp,  
         bupla    TYPE bseg-bupla,  
         secco    TYPE bseg-secco,  
         taxps    TYPE bseg-taxps,  
         gst\_part TYPE bseg-gst\_part,  
         plc\_sup  TYPE bseg-plc\_sup,  
         hsn\_sac  TYPE bseg-hsn\_sac,  
         txgrp    TYPE bseg-txgrp,  
       END OF ty\_bseg,  
       BEGIN OF ty\_adrc1,  
         addrnumber TYPE ad\_addrnum,  
         name1      TYPE ad\_name1,  
         name2      TYPE ad\_name2,  
         city1      TYPE ad\_city1,  
         post\_code1 TYPE ad\_pstcd1,  
         street     TYPE ad\_street,  
         region     TYPE regio,  
       END OF ty\_adrc1.  
  
DATA:gt\_bkpf  TYPE TABLE OF ty\_bkpf,  
     gw\_bkpf  TYPE ty\_bkpf,  
     gt\_bseg  TYPE TABLE OF ty\_bseg,  
     gw\_bseg  TYPE ty\_bseg,  
     gt\_adrc1 TYPE TABLE OF ty\_adrc1,  
     lw\_adrc1 TYPE ty\_adrc1,  
     gt\_adr61 TYPE TABLE OF ty\_adr6,  
     lw\_adr61 TYPE  ty\_adr6.  
CONSTANTS: gc\_ktosl\_joc TYPE ktosl VALUE 'JOC',  
           gc\_ktosl\_jos TYPE ktosl VALUE 'JOS',  
           gc\_ktosl\_tcs TYPE ktosl VALUE 'TCS',  
           gc\_ktosl\_joi TYPE ktosl VALUE 'JOI',  
           gc\_ktosl\_jtc TYPE ktosl VALUE 'JTC',  
           gc\_ktosl\_eGk TYPE ktosl VALUE 'EGK',  
           gc\_shkzg\_h   TYPE shkzg VALUE 'H'.  
DATA:lw\_gstin TYPE ty\_j\_1bbranch,  
     lw\_lfa1  TYPE lfa1.  
\* Data declartion for screen  
DATA: zst\_einv\_api\_struct TYPE ty\_zst\_einv\_api\_struct.  
  
\* Internal table/ varibale declarations declarations  
DATA: gt\_tvarvc           TYPE TABLE OF ty\_tvarvc,  
      gt\_invrefnum        TYPE STANDARD TABLE OF j\_1ig\_invrefnum,  
      gs\_layout           TYPE slis\_layout\_alv,  
      gs\_fcat             TYPE slis\_fieldcat\_alv,  
      gt\_fcat             TYPE STANDARD TABLE OF slis\_fieldcat\_alv,  
      gw\_final            TYPE ty\_final,  
      gt\_final            TYPE TABLE OF ty\_final,  
      gt\_kna1             TYPE TABLE OF ty\_kna1,  
      gt\_adrc             TYPE TABLE OF ty\_adrc,  
      gt\_adr6             TYPE  TABLE OF ty\_adr6,  
      gt\_wb2\_v\_vbrk\_vbrp2 TYPE TABLE OF ty\_wb2\_v\_vbrk\_vbrp2,  
      gt\_wb2\_v\_likp\_lips2 TYPE TABLE OF ty\_wb2\_v\_likp\_lips2,  
      gt\_zteway\_transport TYPE TABLE OF zteway\_transport,  
      gt\_likp\_t           TYPE TABLE OF likp,  
      gw\_likp\_t           TYPE likp,  
      gt\_dfkk             TYPE TABLE OF dfkkbptaxnum,  
      gw\_dfkk             TYPE dfkkbptaxnum,  
      gt\_zzroute          TYPE TABLE OF zzroute,  
      gw\_zzroute          TYPE zzroute,  
      gw\_zteway\_transport TYPE  zteway\_transport,  
      gt\_zteinv\_details   TYPE TABLE OF zteinv\_details,  
      gt\_api              TYPE TABLE OF ty\_api,  
      gt\_appvr            TYPE TABLE OF zteinv\_appvr,  
      gt\_doctyp           TYPE TABLE OF ty\_doctyp,  
      gt\_user\_config      TYPE TABLE OF ty\_user\_config,  
      gt\_wb2\_v\_vbak\_vbap2 TYPE TABLE OF ty\_wb2\_v\_vbak\_vbap2,  
      gt\_vbfa             TYPE TABLE OF ty\_vbfa,  
      gt\_konv             TYPE TABLE OF ty\_konv,  
      gt\_ewaybill         TYPE TABLE OF ty\_ewaybill,  
      gw\_ewaybill         TYPE  ty\_ewaybill,  
      gt\_marc             TYPE TABLE OF ty\_marc,  
      gt\_t005u            TYPE STANDARD TABLE OF ty\_t005u,  
      gt\_doc\_uom          TYPE TABLE OF ty\_doc\_uom,  
      gt\_export           TYPE TABLE OF ty\_export,  
      gt\_lfa1m            TYPE TABLE OF ty\_lfa1,  
      gw\_lfa1             TYPE ty\_lfa1,  
      gt\_bkpfm            TYPE TABLE OF ty\_bkpf1,  
      "gt\_bseg             TYPE TABLE OF ty\_bseg1,  
\*Seller Details  
      gt\_t001w            TYPE TABLE OF ty\_t001w1,  
      gt\_adrc\_s           TYPE TABLE OF ty\_adrc,  
      gt\_adr6\_s           TYPE TABLE OF ty\_adr6,  
      gt\_t005u\_s          TYPE TABLE OF ty\_t005u,  
      gt\_gstin            TYPE TABLE OF ty\_j\_1bbranch,  
      sl\_werks            TYPE RANGE OF vbrp-werks,  
      ss\_werks            LIKE LINE OF sl\_werks,  
      sl\_fkart            TYPE RANGE OF vbrk-fkart,  
      ss\_fkart            LIKE LINE OF sl\_fkart.  
  
DATA: gt\_supply\_values               TYPE TABLE OF   dd07v,  
      gt\_sub\_supply\_values           TYPE TABLE OF   dd07v,  
      gt\_doctyp\_values               TYPE TABLE OF   dd07v,  
      gt\_cancel\_values               TYPE TABLE OF   dd07v,  
      gt\_t\_mode\_values               TYPE TABLE OF   dd07v,  
      gt\_v\_type\_values               TYPE TABLE OF   dd07v,  
      gt\_t\_ext\_reason\_values         TYPE TABLE OF   dd07v,  
      gt\_t\_consignment\_status\_values TYPE TABLE OF   dd07v,  
      gt\_t\_transit\_type\_values       TYPE TABLE OF   dd07v,  
      gt\_v\_reason\_values             TYPE TABLE OF   dd07v.  
  
  
\*Dispatch and Ship Details  
DATA : gt\_likp    TYPE TABLE OF ty\_likp,  
       gt\_kna1\_d  TYPE TABLE OF ty\_kna1,  
       gt\_t005u\_d TYPE TABLE OF ty\_t005u,  
       gt\_adrc\_d  TYPE TABLE OF ty\_adrc,  
       gt\_vbpa    TYPE TABLE OF ty\_vbpa,  
       gt\_lfa1    TYPE TABLE OF lfa1,  
       gt\_makt    TYPE TABLE OF makt,  
       gw\_makt    TYPE makt.  
  
DATA: gt\_einv\_api\_hdr TYPE ztt\_einv\_api\_struct,  
      gt\_einv\_api\_itm TYPE ztt\_einv\_api\_struct\_itm.  
  
DATA: gt\_eway\_api\_hdr TYPE ztt\_eway\_api\_struct,  
      gt\_eway\_api\_itm TYPE ztt\_eway\_api\_struct\_itm.  
  
DATA: gv\_date  TYPE sydatum,  
      gv\_doc   TYPE j\_1ig\_docno,  
      gv\_odn   TYPE xblnr\_alt,  
      gv\_bp    TYPE bupla,  
      gv\_stat  TYPE j\_1ig\_irn\_status,  
      gv\_werks TYPE werks\_d,  
      gv\_docty TYPE zde\_docty,  
      gv\_prctr TYPE prctr.  
  
DATA:gw\_token    TYPE string,  
     gw\_return   TYPE string,  
     ga\_return   TYPE string,  
     gt\_messages TYPE bapiret2\_t,  
     gw\_message  TYPE bapiret2.  
  
DATA: gv\_e\_comm TYPE sy-ucomm.  
  
  
  
DATA: gref\_alv\_container TYPE REF TO cl\_gui\_custom\_container,  
      gref\_alv\_grid      TYPE REF TO cl\_gui\_alv\_grid,  
      ok\_code            LIKE sy-ucomm,  
      gt\_fieldcat        TYPE lvc\_t\_fcat.  
  
DATA:gt\_index TYPE lvc\_t\_row,  
     gt\_row   TYPE lvc\_t\_roid.  
  
  
\* Constant declarations  
CONSTANTS:  
  gc\_sd(2)           TYPE c VALUE 'SD',  
  gc\_mm(2)           TYPE c VALUE 'MM',  
  gc\_fi(2)           TYPE c VALUE 'FI',  
  gc\_sgst(4)         TYPE c VALUE 'JISG',  
  gc\_cgst(4)         TYPE c VALUE 'JICG',  
  gc\_igst(4)         TYPE c VALUE 'JIIG',  
  gc\_kschl\_joig      TYPE kscha VALUE 'JOIG',  
  gc\_kschl\_jocg      TYPE kscha VALUE 'JOCG',  
  gc\_kschl\_josg      TYPE kscha VALUE 'JOSG',  
  gc\_kschl\_joug      TYPE kscha VALUE 'JOUG',  
  gc\_kschl\_cess      TYPE kscha VALUE 'JOCC',  
  gc\_kschl\_jwts      TYPE kscha VALUE 'JWTS',  
  gc\_kschl\_jtc1      TYPE kscha VALUE 'JTC1',  
  gc\_kschl\_zy19      TYPE kscha VALUE 'ZY19',  
  gc\_kschl\_zr00      TYPE kscha VALUE 'ZR00',  
  gc\_sign(1)         TYPE c VALUE 'I',  
  gc\_equal(2)        TYPE c VALUE 'EQ',  
  gc\_einv            TYPE ui\_func VALUE 'EINV',  
  gc\_eway            TYPE ui\_func VALUE 'EWAY',  
  gc\_edit            TYPE ui\_func VALUE 'EWAYT',  
  gc\_btn\_type\_2      TYPE tb\_btype VALUE '2',  
  gc\_btn\_type\_0      TYPE tb\_btype VALUE '0',  
  gc\_icon\_einv       TYPE iconname VALUE '@UO@',  
  gc\_icon\_eway       TYPE iconname VALUE '@0K@',  
  gc\_icon\_edit       TYPE iconname VALUE '@HL@',  
  gc\_icon\_39         TYPE iconname VALUE '@39@',  
  gc\_icon\_jz         TYPE iconname VALUE '@JZ@',  
  gc\_icon\_08         TYPE iconname VALUE '@08@',  
  gc\_icon\_b2         TYPE iconname VALUE '@B2@',  
  gc\_icon\_0a         TYPE iconname VALUE '@0A@',  
  gc\_icon\_0w         TYPE iconname VALUE '@0W@',  
  gc\_ucomm\_einv      TYPE sy-ucomm VALUE 'EINV',  
  gc\_ucomm\_eway      TYPE sy-ucomm VALUE 'EWAY',  
  gc\_ucomm\_einvg     TYPE sy-ucomm VALUE 'EINVG',  
  gc\_ucomm\_einvc     TYPE sy-ucomm VALUE 'EINVC',  
  gc\_ucomm\_einvi     TYPE sy-ucomm VALUE 'EINVI',  
  gc\_ucomm\_eeayg     TYPE sy-ucomm VALUE 'EEAYG',  
  gc\_ucomm\_eeayb     TYPE sy-ucomm VALUE 'EEAYB',  
  gc\_ucomm\_ewayu     TYPE sy-ucomm VALUE 'EWAYU',  
  gc\_ucomm\_ewayv     TYPE sy-ucomm VALUE 'EWAYV',  
  gc\_ucomm\_ewaye     TYPE sy-ucomm VALUE 'EWAYE',  
  gc\_ucomm\_ewayc     TYPE sy-ucomm VALUE 'EWAYC',  
  gc\_ucomm\_ewayp     TYPE sy-ucomm VALUE 'EWAYP',  
  gc\_ucomm\_ewayt     TYPE sy-ucomm VALUE 'EWAYT',  
  gc\_fcode\_einvg     TYPE ui\_func VALUE 'EINVG',  
  gc\_fcode\_einvc     TYPE ui\_func VALUE 'EINVC',  
  gc\_fcode\_eeayg     TYPE ui\_func VALUE 'EEAYG',  
  gc\_fcode\_eeayb     TYPE ui\_func VALUE 'EEAYB',  
  gc\_fcode\_ewayp     TYPE ui\_func VALUE 'EWAYP',  
  gc\_fcode\_ewayu     TYPE ui\_func VALUE 'EWAYU',  
  gc\_fcode\_ewayv     TYPE ui\_func VALUE 'EWAYV',  
  gc\_fcode\_ewaye     TYPE ui\_func VALUE 'EWAYE',  
  gc\_fcode\_ewayc     TYPE ui\_func VALUE 'EWAYC',  
  gc\_land1\_in        TYPE land1 VALUE 'IN',  
  gc\_vbtyp\_v         TYPE vbtyp\_v VALUE 'C',  
  gc\_spras\_en        TYPE spras VALUE 'EN',  
  gc\_apiid\_token     TYPE zde\_apiid VALUE 'ACC\_TOKEN',  
  gc\_irn\_sts\_act     TYPE j\_1ig\_irn\_status VALUE 'ACT',  
  gc\_irn\_sts\_err     TYPE j\_1ig\_irn\_status VALUE 'ERR',  
  gc\_irn\_sts\_cnl     TYPE j\_1ig\_irn\_status VALUE 'CNL',  
  gc\_eway\_sts\_a      TYPE j\_1ig\_stat VALUE 'A',  
  gc\_eway\_sts\_e      TYPE j\_1ig\_stat VALUE 'E',  
  gc\_eway\_sts\_c      TYPE j\_1ig\_stat VALUE 'C',  
  gc\_v\_type\_1        TYPE zde\_v\_type VALUE '1',  
  gc\_periv\_v3        TYPE periv VALUE 'V3',  
  gc\_parvw\_we        TYPE parvw VALUE 'WE',  
  gc\_parvw\_re        TYPE parvw VALUE 'RE',  
  gc\_lay\_mode\_a      TYPE lvc\_libox VALUE 'A',  
  gc\_alv\_save\_a      TYPE char01 VALUE 'A',  
  gc\_sup\_type\_b2b    TYPE char10 VALUE '"B2B"',  
  gc\_sup\_type\_expwp  TYPE char10 VALUE '"EXPWP"',  
  gc\_sup\_type\_expwop TYPE char10 VALUE '"EXPWOP"',  
  gc\_charge\_y        TYPE char10 VALUE '"Y"',  
  gc\_charge\_n        TYPE char10 VALUE '"N"',  
  gc\_docty\_inv       TYPE char15 VALUE 'INV',  
  gc\_apiid\_usrgstin  TYPE  zde\_apiid VALUE 'USRGSTIN',  
  gc\_apiid\_eusrgstin TYPE  zde\_apiid VALUE 'E\_USRGSTIN',  
  gc\_apiid\_esupgstin TYPE  zde\_apiid VALUE 'E\_SUPGSTIN',  
  gc\_apiid\_ebuygstin TYPE  zde\_apiid VALUE 'E\_BUYGSTIN',  
  gc\_gstin\_urp       TYPE char20 VALUE '"URP"',  
  gc\_export\_place    TYPE char100 VALUE '"OTHER COUNTRY"',  
  gc\_exp\_state       TYPE char100 VALUE '96',  
  gc\_ttype\_shg       TYPE char10 VALUE '"SHG"',  
  gc\_ttype\_reg       TYPE char10 VALUE '"REG"',  
  gc\_tdid\_0002       TYPE tdid VALUE  '0002',  
  gc\_tobject\_vbbk    TYPE tdobject VALUE 'VBBK',  
  gc\_service\_y       TYPE char5 VALUE '"Y"',  
  gc\_service\_n       TYPE char5 VALUE '"N"',  
  gc\_pstyv\_tad       TYPE pstyv VALUE 'TAD',  
  gc\_attr\_einv       TYPE char100 VALUE '"EINV"',  
  gc\_msgid\_01        TYPE sy-msgid VALUE '01',  
  gc\_msgno\_319       TYPE sy-msgno VALUE '319',  
  gc\_mtype\_s         TYPE bapi\_mtype VALUE 'S',  
  gc\_mtype\_e         TYPE bapi\_mtype VALUE 'E',  
  gc\_tmode\_5         TYPE zde\_t\_mode VALUE '5',  
  gc\_sub\_suptyp\_exp  TYPE char30 VALUE   '"EXPORT"',  
  gc\_gen\_sts\_1       TYPE char15 VALUE '"1"',  
  gc\_dsource\_erp     TYPE char20 VALUE '"ERP"',  
  gc\_ewaysts\_abc     TYPE char5 VALUE '"ABC"',  
  gc\_ewaysts\_ac      TYPE char5 VALUE '"AC"',  
  gc\_eway\_print      TYPE char5 VALUE '"N"',  
  gc\_okcode\_furt     TYPE char10 VALUE 'FURT',  
  gc\_fname\_crec\_code TYPE fieldname VALUE 'C\_REASON\_CODE',  
  gc\_fname\_crec      TYPE fieldname VALUE 'C\_REASON',  
  gc\_fname\_erec\_code TYPE fieldname VALUE 'E\_REASON\_CODE',  
  gc\_fname\_erec      TYPE fieldname VALUE 'E\_REASON',  
  gc\_form\_eway       TYPE char30 VALUE 'USER\_COMMAND\_EWAY\_CANCEL',  
  gc\_form\_einv       TYPE char30 VALUE 'USER\_COMMAND\_EINV\_CANCEL',  
  gc\_tname\_einv      TYPE tabname VALUE 'ZTEINV\_DETAILS',  
  gc\_tname\_eway      TYPE tabname VALUE 'ZTEWAY\_TRANSPORT',  
  gc\_curr\_inr        TYPE waers  VALUE 'INR',  
  gc\_var\_unit        TYPE rvari\_vnam VALUE 'ZEDOC\_UNIT',  
  gc\_var\_disc        TYPE rvari\_vnam VALUE 'ZEDOC\_DISC'.

Include ZREP\_EINV\_EWAY\_REPORT\_CLASS

\*&---------------------------------------------------------------------\*  
\*&  Include           ZREP\_EINV\_EWAY\_REPORT\_CLASS  
\*&---------------------------------------------------------------------\*  
  
\* Data declartion for class  
  
      DATA:lw\_tool TYPE stb\_button,  
           lw\_menu TYPE stb\_btnmnu.  
      DATA:dg\_dyndoc\_id   TYPE REF TO cl\_dd\_document,  
           dg\_splitter    TYPE REF TO cl\_gui\_splitter\_container,  
           dg\_parent\_grid TYPE REF TO cl\_gui\_container,  
           dg\_html\_cntrl  TYPE REF TO cl\_gui\_html\_viewer,  
           dg\_parent\_html TYPE REF TO cl\_gui\_container.  
  
  
\* Class defenition  
      CLASS lcl\_event DEFINITION .  
  
        PUBLIC SECTION.  
          METHODS :toolbar FOR EVENT toolbar OF  cl\_gui\_alv\_grid  
                       IMPORTING e\_object,  
                   user\_command FOR EVENT user\_command OF cl\_gui\_alv\_grid  
                       IMPORTING e\_ucomm,  
                   menu FOR EVENT menu\_button  OF cl\_gui\_alv\_grid  
                       IMPORTING e\_object  
                                 e\_ucomm ,  
                   handle\_print\_top\_of\_page  
                       FOR EVENT print\_top\_of\_page OF  
                                      cl\_gui\_alv\_grid,  
                   handle\_top\_of\_page FOR EVENT top\_of\_page OF  
                       cl\_gui\_alv\_grid IMPORTING e\_dyndoc\_id,  
  
                   handle\_hotspot\_click FOR EVENT hotspot\_click OF  
                       cl\_gui\_alv\_grid IMPORTING e\_row\_id e\_column\_id.  
      ENDCLASS.  
  
  
  
      DATA: go\_obj TYPE REF TO lcl\_event.  
  
  
  
\* Class Implementation  
      CLASS lcl\_event IMPLEMENTATION.  
  
\* Toolbar  
        METHOD toolbar.  
  
          READ TABLE gt\_doctyp TRANSPORTING NO FIELDS WITH KEY einv = abap\_true.  
          IF sy-subrc IS INITIAL.  
            lw\_tool-function = gc\_einv.  
            lw\_tool-text     = 'E-Invoice Generation'(040).  
            lw\_tool-butn\_type = gc\_btn\_type\_2.  
            lw\_tool-icon     = gc\_icon\_einv.  
            APPEND lw\_tool TO e\_object->mt\_toolbar.  
          ENDIF.  
  
          READ TABLE gt\_doctyp TRANSPORTING NO FIELDS WITH KEY eway = abap\_true.  
          IF sy-subrc IS INITIAL.  
  
            lw\_tool-function = gc\_eway.  
            lw\_tool-text     = 'E-Way Bill Generation'(041).  
            lw\_tool-butn\_type = gc\_btn\_type\_2.  
            lw\_tool-icon     = gc\_icon\_eway.  
            APPEND lw\_tool TO e\_object->mt\_toolbar.  
  
            lw\_tool-function = gc\_edit.  
            lw\_tool-text     = 'Edit Transporter (SAP Only)'(042).  
            lw\_tool-icon     = gc\_icon\_edit.  
            lw\_tool-butn\_type = gc\_btn\_type\_0.  
            APPEND lw\_tool TO e\_object->mt\_toolbar.  
  
          ENDIF.  
        ENDMETHOD.             "DISPLAY  
  
\* Menu  
  
        METHOD menu .  
  
          IF e\_ucomm = gc\_ucomm\_einv.  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_einvg  
                text  = 'Generate E-Invoice'(043).  
\*                icon  = gc\_icon\_39.  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_einvc  
                text  = 'Cancel E-Invoice'(044).  
  
            .  
          ENDIF.  
  
          IF e\_ucomm = gc\_ucomm\_eway.  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_eeayg  
                text  = 'Generate E-Way Bill(Part A & B)'.  
  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_eeayb  
                text  = 'Generate E-Way Bill(Part A)'.  
  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_ewayp  
                text  = 'Download E-Way Bill'.  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_ewayu  
                text  = 'Update E-Way Bill Transporter'.  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_ewayv  
                text  = 'Update E-Way Bill Vehicle No.'.  
  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_ewaye  
                text  = 'Extend E-Way Bill Validity'.  
            .  
            CALL METHOD e\_object->add\_function  
              EXPORTING  
                fcode = gc\_fcode\_ewayc  
                text  = 'Cancel E-Way Bill'.  
  
  
          ENDIF.  
  
  
        ENDMETHOD.  
  
  
\* User Command  
        METHOD user\_command.  
  
          DATA: lw\_index TYPE lvc\_s\_row,  
                lv\_val   TYPE xfeld.  
  
  
          gv\_e\_comm = e\_ucomm.  
  
          CASE e\_ucomm.  
            WHEN gc\_ucomm\_einvg.  
              PERFORM einvoice\_generate.  
  
            WHEN gc\_ucomm\_einvc.  
              PERFORM einvoice\_cancel\_old.  
  
            WHEN gc\_ucomm\_eeayg  OR gc\_ucomm\_eeayb.  
              PERFORM ewaybill\_generate.  
  
            WHEN gc\_ucomm\_ewayu.  
              PERFORM ewaybill\_update\_transporter.  
  
            WHEN gc\_ucomm\_ewayv.  
              PERFORM ewaybill\_update\_vehicle.  
  
            WHEN gc\_ucomm\_ewaye.  
              PERFORM ewaybill\_extend.  
  
            WHEN gc\_ucomm\_ewayc.  
              PERFORM ewaybill\_cancel\_old.  
  
            WHEN gc\_ucomm\_ewayp.  
              PERFORM ewaybill\_print.  
  
            WHEN gc\_ucomm\_ewayt.  
              PERFORM check\_selected\_data CHANGING lv\_val.  
              IF lv\_val IS INITIAL.  
                CALL SCREEN 9001 STARTING AT 20 20 ENDING AT 120 75.  
              ELSE.  
                MESSAGE 'Please select at least one document to edit Transporter details'(007) TYPE 'I'.  
                EXIT.  
              ENDIF.  
  
          ENDCASE.  
  
        ENDMETHOD.                    "USER\_COMMAND  
\* Top-of-Page  
  
        METHOD handle\_print\_top\_of\_page.  
          IF sy-pagno = 1.  
          ENDIF.  
        ENDMETHOD.  
  
\* Handle Top-of-page  
        METHOD handle\_top\_of\_page.  
          PERFORM top\_of\_page USING dg\_dyndoc\_id.  
        ENDMETHOD.  
  
  
\* Hotsspot click  
  
        METHOD  handle\_hotspot\_click.  
          IF e\_column\_id = 'VBELN'.  
            PERFORM handle\_hotspot\_on\_vbeln USING e\_row\_id.   " Subroutine to handle hotspot on customer number  
          ENDIF.  
        ENDMETHOD.  
  
  
      ENDCLASS.                    "LCL\_EVENT IMPLEMENTATION  
\*&---------------------------------------------------------------------\*  
\*&      Form  TOP\_OF\_PAGE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
      FORM top\_of\_page USING   dg\_dyndoc\_id TYPE REF TO cl\_dd\_document.  
        DATA : dl\_text(255) TYPE c,  
               lv\_line      TYPE i,  
               lv\_lines     TYPE char10.  
\* Populating header to top-of-page  
        READ TABLE gt\_final INTO gw\_final INDEX 1.  
        CALL METHOD dg\_dyndoc\_id->add\_text  
          EXPORTING  
            text      = 'E-Invoice/E-Way Bill Register'(052)  
            sap\_style = cl\_dd\_area=>heading.  
  
        CALL METHOD dg\_dyndoc\_id->new\_line.  
        CLEAR : dl\_text.  
        CONCATENATE 'Company Code:'(053) gw\_final-bukrs  
               INTO dl\_text SEPARATED BY space.  
        PERFORM add\_text USING dl\_text.  
\* Add new-line  
        CALL METHOD dg\_dyndoc\_id->new\_line.  
  
  
        CLEAR : dl\_text.  
        CONCATENATE 'Generator/Approver:'(054) sy-uname  
              INTO dl\_text SEPARATED BY space.  
        PERFORM add\_text USING dl\_text.  
  
        CALL METHOD dg\_dyndoc\_id->new\_line.  
  
  
        CLEAR : dl\_text.  
        DESCRIBE TABLE  gt\_final LINES lv\_line.  
        lv\_lines = lv\_line.  
  
        CONCATENATE 'No of Record(s):'(055) lv\_lines  
              INTO dl\_text SEPARATED BY space.  
        PERFORM add\_text USING dl\_text.  
  
        CALL METHOD dg\_dyndoc\_id->new\_line.  
  
  
\* Populating data to html control  
        PERFORM html.  
      ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  ADD\_TEXT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_DL\_TEXT  text  
\*----------------------------------------------------------------------\*  
      FORM add\_text USING p\_text TYPE sdydo\_text\_element.  
\* Adding text  
        CALL METHOD dg\_dyndoc\_id->add\_text  
          EXPORTING  
            text         = p\_text  
            sap\_emphasis = cl\_dd\_area=>heading.  
      ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  HTML  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
      FORM html .  
        DATA : dl\_length        TYPE i,                           " Length  
               dl\_background\_id TYPE sdydo\_key VALUE space. " Background\_id  
\* Creating html control  
        IF dg\_html\_cntrl IS INITIAL.  
          CREATE OBJECT dg\_html\_cntrl  
            EXPORTING  
              parent = dg\_parent\_html.  
        ENDIF.  
\* Reuse\_alv\_grid\_commentary\_set  
        CALL FUNCTION 'REUSE\_ALV\_GRID\_COMMENTARY\_SET'  
          EXPORTING  
            document = dg\_dyndoc\_id  
            bottom   = space  
          IMPORTING  
            length   = dl\_length.  
\* Get TOP->HTML\_TABLE ready  
        CALL METHOD dg\_dyndoc\_id->merge\_document.  
\* Set wallpaper  
        CALL METHOD dg\_dyndoc\_id->set\_document\_background  
          EXPORTING  
            picture\_id = dl\_background\_id.  
\* Connect TOP document to HTML-Control  
        dg\_dyndoc\_id->html\_control = dg\_html\_cntrl.  
\* Display TOP document  
        CALL METHOD dg\_dyndoc\_id->display\_document  
          EXPORTING  
            reuse\_control      = 'X'  
            parent             = dg\_parent\_html  
          EXCEPTIONS  
            html\_display\_error = 1.  
        IF sy-subrc NE 0.  
          MESSAGE  'Error in displaying top-of-page'(036) TYPE 'I'.  
        ENDIF.  
      ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  HANDLE\_HOTSPOT\_ON\_VBELN  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_E\_ROW\_ID  text  
\*----------------------------------------------------------------------\*  
      FORM handle\_hotspot\_on\_vbeln  USING    p\_e\_row\_id.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  HANDLE\_HOTSPOT\_ON\_VBELN  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_E\_ROW\_ID  text  
\*----------------------------------------------------------------------\*  
  
        DATA: lw\_final TYPE ty\_final.  
  
        READ TABLE gt\_final INTO lw\_final INDEX p\_e\_row\_id.  
        IF sy-subrc IS INITIAL.  
          IF  lw\_final-mod = gc\_sd.  
  
            CALL FUNCTION 'AUTHORITY\_CHECK\_TCODE'  
              EXPORTING  
                tcode  = 'VF03'  
              EXCEPTIONS  
                ok     = 1  
                not\_ok = 2  
                OTHERS = 3.  
            IF sy-subrc <> 1.  
              MESSAGE 'You are not a authorised User for VF03'(056) TYPE 'I'.  
              EXIT.  
            ELSEIF sy-subrc = 1.  
              SET PARAMETER ID 'VF' FIELD lw\_final-vbeln.  
              CALL TRANSACTION  'VF03' AND SKIP FIRST SCREEN.  
            ENDIF.  
          ELSEif lw\_final-mod = gc\_fi.  
            CALL FUNCTION 'AUTHORITY\_CHECK\_TCODE'  
              EXPORTING  
                tcode  = 'FB03'  
              EXCEPTIONS  
                ok     = 1  
                not\_ok = 2  
                OTHERS = 3.  
            IF sy-subrc <> 1.  
              MESSAGE 'You are not a authorised User for FB03'(056) TYPE 'I'.  
              EXIT.  
            ELSEIF sy-subrc = 1.  
              SET PARAMETER ID 'BLN' FIELD lw\_final-vbeln.  
              SET PARAMETER ID 'BUK' FIELD lw\_final-bukrs.  
              SET PARAMETER ID 'GJR' FIELD lw\_final-gjahr.  
              CALL TRANSACTION  'FB03' AND SKIP FIRST SCREEN.  
            ENDIF.  
          ENDIF.  
  
        ENDIF.  
      ENDFORM.

INCLUDE zrep\_einv\_eway\_report\_sel.

\*&---------------------------------------------------------------------\*  
\*&  Include           ZREP\_EINV\_EWAY\_REPORT\_SEL  
\*&---------------------------------------------------------------------\*  
  
SELECTION-SCREEN BEGIN OF BLOCK b1 WITH FRAME TITLE TEXT-t01.  .  
PARAMETERS: p\_ccode   TYPE bukrs OBLIGATORY,  
            p\_fyear   TYPE gjahr OBLIGATORY,  
            p\_mod(10) AS LISTBOX VISIBLE LENGTH 30 OBLIGATORY.  
  
SELECT-OPTIONS:  
                s\_werks FOR gv\_werks NO INTERVALS,  
                s\_prctr FOR gv\_prctr MODIF ID m1,  
                s\_docty FOR gv\_docty NO INTERVALS,  
                s\_doc   FOR gv\_doc,  
                s\_date  FOR gv\_date OBLIGATORY.  
  
SELECTION-SCREEN END OF BLOCK b1.  
  
  
  
SELECTION-SCREEN BEGIN OF BLOCK b2 WITH FRAME TITLE TEXT-t04.  .  
\*Usage guidlines comment:  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 1(5) icon.  
SELECTION-SCREEN COMMENT 6(50) TEXT-t02.  
SELECTION-SCREEN END OF LINE.  
  
\*E-Invoice comment:  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 1(5) icon1.  
SELECTION-SCREEN COMMENT 6(50) TEXT-t03.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t05.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t06.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t09.  
SELECTION-SCREEN END OF LINE.  
  
\*E-Way Bill Comment:  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 1(5) icon2.  
SELECTION-SCREEN COMMENT 6(50) TEXT-t10.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t11.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t12.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t13.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t14.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t15.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(50) TEXT-t09.  
SELECTION-SCREEN END OF LINE.  
  
\*Pre-requisites Comment:  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 1(5) icon3.  
SELECTION-SCREEN COMMENT 6(50) TEXT-t16.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(79) TEXT-t17.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(79) TEXT-t18.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN COMMENT 8(79) TEXT-t19.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN END OF BLOCK b2.  
  
  
AT SELECTION-SCREEN .  
  
AT SELECTION-SCREEN ON p\_ccode.  
  
  IF gt\_user\_config IS INITIAL.  
    MESSAGE 'E-Invooice/E-way bill configuration is empty for your login'(000) TYPE 'E'.  
  ELSE.  
    READ TABLE gt\_user\_config TRANSPORTING NO FIELDS WITH KEY bukrs = p\_ccode.  
    IF sy-subrc IS NOT INITIAL.  
      MESSAGE 'You are not authorized for given company code'(001) TYPE 'E'.  
    ENDIF.  
  ENDIF.  
  
AT SELECTION-SCREEN ON p\_mod.  
  IF gt\_user\_config IS INITIAL.  
    MESSAGE 'E-Invooice/E-way bill configuration is empty for your login'(002) TYPE 'E'.  
  ELSE.  
    READ TABLE gt\_user\_config TRANSPORTING NO FIELDS WITH KEY zmodule = p\_mod.  
    IF sy-subrc IS NOT INITIAL.  
      MESSAGE 'You are not authorized for given Module'(003) TYPE 'E'.  
    ENDIF.  
  ENDIF.  
  
  
AT SELECTION-SCREEN ON s\_werks.  
  IF gt\_user\_config IS INITIAL.  
    MESSAGE 'E-Invooice/E-way bill configuration is empty for your login'(004) TYPE 'E'.  
  ELSE.  
  
    LOOP AT s\_werks.  
      READ TABLE gt\_user\_config TRANSPORTING NO FIELDS WITH KEY werks = s\_werks-low.  
      IF sy-subrc IS NOT INITIAL.  
        MESSAGE 'You are not authorized for given Plant'(005) TYPE 'E'.  
      ENDIF.  
    ENDLOOP.  
  
  ENDIF.  
  
AT SELECTION-SCREEN ON s\_docty.  
  IF gt\_user\_config IS INITIAL.  
    MESSAGE 'E-Invooice/E-way bill configuration is empty for your login'(006) TYPE 'E'.  
  ELSE.  
    LOOP AT s\_docty.  
      READ TABLE gt\_user\_config TRANSPORTING NO FIELDS WITH KEY fkart = s\_docty-low.  
      IF sy-subrc IS NOT INITIAL.  
        MESSAGE 'You are not authorized for given Document Type'(011) TYPE 'E'.  
      ENDIF.  
    ENDLOOP.  
  
  ENDIF.

INCLUDE zrep\_einv\_eway\_report\_forms.

\*&---------------------------------------------------------------------\*  
\*&  Include           ZREP\_EINV\_EWAY\_REPORT\_FORMS  
\*&---------------------------------------------------------------------\*  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_data .  
  
  DATA:lw\_user\_config TYPE ty\_user\_config.  
  
\* Get document related configuration data  
  SELECT bukrs  
        zmodule  
        fkart  
        vtext  
        sup\_type  
        sub\_type  
        edoc\_type  
        einv  
        eway  
        accpost  
        senderr  
        instgen  
        batch   FROM zteinv\_doctyp  
           INTO  TABLE gt\_doctyp  
           WHERE bukrs EQ p\_ccode AND  
                 zmodule EQ p\_mod AND  
                 fkart IN s\_docty AND  
                 batch EQ abap\_true.  
  IF sy-subrc IS NOT INITIAL.  
    MESSAGE 'Batch generation config is empty for the given selection'(008) TYPE 'E'.  
  ENDIF.  
  
  
\* Get generator/Approver data  
  SELECT bukrs  
        zmodule  
        fkart  
        werks  
        prctr  
        gen\_apr  
        blk\_usr   FROM zteinv\_appvr  
           INTO TABLE gt\_user\_config  
           WHERE bukrs EQ p\_ccode AND  
                 zmodule EQ p\_mod AND  
                 fkart IN s\_docty AND  
                 werks IN s\_werks AND  
                 prctr IN s\_prctr AND  
                 gen\_apr = sy-uname.  
  IF sy-subrc IS NOT INITIAL.  
    MESSAGE 'Config is empty for the given selection'(009) TYPE 'E'.  
  ENDIF.  
  
\* get the API data  
  SELECT apiid apiprov FROM zteinv\_api INTO TABLE gt\_api.  
  
  IF p\_mod = gc\_sd.  
  
\* collect the billing type and plant details  
    CLEAR:sl\_fkart,sl\_werks.  
    IF gt\_user\_config IS NOT INITIAL.  
      LOOP AT gt\_user\_config INTO lw\_user\_config  WHERE zmodule = gc\_sd  
                                                  AND fkart IN s\_docty.  
        ss\_fkart-sign = gc\_sign.  
        ss\_fkart-option = gc\_equal.  
        ss\_fkart-low = lw\_user\_config-fkart.  
        APPEND ss\_fkart TO sl\_fkart.  
        CLEAR:ss\_fkart.  
  
        ss\_werks-sign = gc\_sign.  
        ss\_werks-option = gc\_equal.  
        ss\_werks-low = lw\_user\_config-werks.  
        APPEND ss\_werks TO sl\_werks.  
        CLEAR ss\_werks.  
      ENDLOOP.  
      SORT sl\_fkart.  
      DELETE ADJACENT DUPLICATES FROM sl\_fkart COMPARING ALL FIELDS.  
      SORT sl\_werks.  
      DELETE ADJACENT DUPLICATES FROM sl\_werks COMPARING ALL FIELDS.  
    ENDIF.  
  
  
    PERFORM process\_sd\_data.  
    PERFORM process\_sd\_data\_delivery.  
  ELSEIF p\_mod = gc\_fi.  
  
\* collect the billing type and plant details  
    CLEAR:sl\_fkart,sl\_werks.  
    IF gt\_user\_config IS NOT INITIAL.  
      LOOP AT gt\_user\_config INTO lw\_user\_config  WHERE zmodule = gc\_fi  
                                                  AND fkart IN s\_docty.  
        ss\_fkart-sign = gc\_sign.  
        ss\_fkart-option = gc\_equal.  
        ss\_fkart-low = lw\_user\_config-fkart.  
        APPEND ss\_fkart TO sl\_fkart.  
        CLEAR:ss\_fkart.  
  
        ss\_werks-sign = gc\_sign.  
        ss\_werks-option = gc\_equal.  
        ss\_werks-low = lw\_user\_config-werks.  
        APPEND ss\_werks TO sl\_werks.  
        CLEAR ss\_werks.  
      ENDLOOP.  
      SORT sl\_fkart.  
      DELETE ADJACENT DUPLICATES FROM sl\_fkart COMPARING ALL FIELDS.  
      SORT sl\_werks.  
      DELETE ADJACENT DUPLICATES FROM sl\_werks COMPARING ALL FIELDS.  
    ENDIF.  
  
    PERFORM process\_fi\_data.  
  
  ENDIF.  
  PERFORM change\_region\_text. "Added by Lakshmi  
  
ENDFORM.  
  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  SET\_MODULE\_VALUE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM set\_module\_value .  
  
  DATA:lt\_values    TYPE TABLE OF   dd07v,  
       ls\_value     TYPE dd07v,  
       lv\_name      TYPE vrm\_id,  
       lt\_list      TYPE vrm\_values,  
       lw\_mod\_value LIKE LINE OF lt\_list.  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_ZMODULE'  
    TABLES  
      values\_tab      = lt\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
    LOOP AT lt\_values INTO ls\_value.  
      lw\_mod\_value-key =  ls\_value-domvalue\_l.  
      lw\_mod\_value-text = ls\_value-ddtext.  
      APPEND lw\_mod\_value TO lt\_list.  
    ENDLOOP.  
  
    lv\_name = 'P\_MOD'.  
    CALL FUNCTION 'VRM\_SET\_VALUES'  
      EXPORTING  
        id     = lv\_name  
        values = lt\_list.  
  ENDIF.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  FETCH\_USER\_CONFIG  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM fetch\_user\_config .  
  
  DATA:lw\_token    TYPE string,  
       lw\_return   TYPE string,  
       wa\_return   TYPE string,  
       lt\_messages TYPE bapiret2\_t,  
       lw\_message  TYPE bapiret2.  
  
\* get approver details  
  SELECT bukrs  
         zmodule  
         fkart  
         werks  
         prctr  
         gen\_apr  
         blk\_usr   FROM zteinv\_appvr INTO TABLE gt\_user\_config WHERE gen\_apr = sy-uname AND  
                                                             blk\_usr EQ ''.  
  IF sy-subrc IS NOT INITIAL.  
\* Through error message  
    EXIT.  
  ENDIF.  
  
\*\*\* Fetch UOM Mapping  
  SELECT edoc\_uom meins FROM ztdoc\_uom INTO TABLE gt\_doc\_uom WHERE edoc\_uom IS NOT NULL.  
  
\* Get export port details.  
  SELECT zport\_code  
          zport  
          land1  
          zport\_state  
          zport\_address1  
          zport\_address2  
          zport\_place  
          zport\_pincode FROM ztedoc\_export INTO TABLE gt\_export WHERE land1 = gc\_land1\_in.  
  
\*\*\* generate token  
  CLEAR:gw\_token,lw\_return.  
  SELECT SINGLE apiuri FROM zteinv\_api INTO gw\_token WHERE apiid = gc\_apiid\_token.  
  IF sy-subrc IS NOT INITIAL.  
    IF gw\_token IS INITIAL.  
      CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
        IMPORTING  
          ex\_token    = gw\_token  
          ex\_return   = lw\_return  
          et\_messages = lt\_messages.  
    ENDIF.  
  ENDIF.  
ENDFORM.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  SD\_PROCESS\_DATA  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM process\_sd\_data .  
  
  
  DATA: lw\_invrefnum        TYPE j\_1ig\_invrefnum,  
        lw\_wb2\_v\_vbrk\_vbrp2 TYPE ty\_wb2\_v\_vbrk\_vbrp2,  
        lw\_kna1             LIKE LINE OF gt\_kna1,  
        lw\_gstin            LIKE LINE OF gt\_gstin,  
        lv\_internal\_date    TYPE sy-datum,  
        lw\_konv             TYPE ty\_konv,  
        lw\_ewaybill         TYPE ty\_ewaybill,  
        ls\_zteway\_transport TYPE zteway\_transport,  
        ls\_zteinv\_details   TYPE zteinv\_details,  
        lw\_vbpa             TYPE ty\_vbpa,  
        lw\_kna1\_d           TYPE ty\_kna1,  
        lw\_lfa1             TYPE lfa1,  
        lt\_likp             TYPE TABLE OF ty\_likp,  
        lv\_gjahr(4)         TYPE c,  
        lt\_vbrk\_vbrp        TYPE TABLE OF ty\_wb2\_v\_vbrk\_vbrp2,  
        lw\_vbrk\_vbrp2       TYPE ty\_wb2\_v\_vbrk\_vbrp2,  
        lw\_doctyp           TYPE ty\_doctyp,  
        lt\_sd\_header        TYPE TABLE OF ty\_wb2\_v\_vbrk\_vbrp2,  
        lt\_vbrk\_export      TYPE TABLE OF ty\_wb2\_v\_vbrk\_vbrp2,  
        lv\_index            TYPE sy-tabix,  
        lv\_exp\_curr         TYPE xfeld.  
  
  DATA:rt\_name TYPE RANGE OF rvari\_vnam,  
       rs\_name LIKE LINE OF rt\_name.  
  
\*& Get Invocies for E-invocie and E-way bill  
  IF s\_prctr IS NOT INITIAL.  
  
  
    SELECT vbeln  vbeln\_i posnr\_i fkart fktyp vbtyp waerk vkorg  
           vtweg  knumv fkdat gjahr land1 bukrs netwr ernam kunrg  
           kunag xblnr zuonr fksto bupla fkimg\_i  meins\_i kursk\_i  
           netwr\_i vgbel\_i  matnr\_i arktx\_i charg\_i pstyv\_i werks\_i prctr\_i mwsbp\_i  
           FROM wb2\_v\_vbrk\_vbrp2  
             INTO TABLE gt\_wb2\_v\_vbrk\_vbrp2  
             WHERE fkart IN sl\_fkart  
             AND vbeln IN s\_doc  
             AND fkdat IN s\_date  
             AND bukrs = p\_ccode  
             AND werks\_i  IN sl\_werks  
             AND prctr\_i IN s\_prctr.  
  
  ELSE.  
  
  
    SELECT vbeln  vbeln\_i posnr\_i fkart fktyp vbtyp waerk vkorg  
           vtweg  knumv fkdat gjahr land1 bukrs netwr ernam kunrg  
           kunag xblnr zuonr fksto bupla fkimg\_i  meins\_i kursk\_i  
           netwr\_i vgbel\_i  matnr\_i arktx\_i charg\_i pstyv\_i werks\_i prctr\_i mwsbp\_i  
       FROM wb2\_v\_vbrk\_vbrp2  
             INTO TABLE gt\_wb2\_v\_vbrk\_vbrp2  
             WHERE fkart IN sl\_fkart  
             AND vbeln IN s\_doc  
             AND fkdat IN s\_date  
             AND bukrs = p\_ccode  
             AND werks\_i  IN sl\_werks.  
  
  ENDIF.  
  
  
  
  
  IF gt\_wb2\_v\_vbrk\_vbrp2 IS NOT INITIAL.  
  
    lt\_sd\_header = gt\_wb2\_v\_vbrk\_vbrp2.  
  
    SORT lt\_sd\_header BY bukrs fkart vbeln.  
    DELETE ADJACENT DUPLICATES FROM lt\_sd\_header COMPARING  bukrs fkart vbeln.  
  
    IF lt\_sd\_header IS NOT INITIAL.  
\* Condition value  
      SELECT knumv kposn  kschl kbetr kwert FROM prcd\_elements"konv "Changed by Lakshmi  
                           INTO TABLE gt\_konv  
                           FOR ALL ENTRIES IN lt\_sd\_header  
                           WHERE knumv = lt\_sd\_header-knumv  
                             AND kinak = abap\_false.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_konv BY knumv kposn kschl.  
      ENDIF.  
  
\* Eway bill  
      SELECT bukrs  
            doctyp  
            docno  
            gjahr  
            ebillno  
            egen\_dat  
            egen\_time  
            vdfmdate  
            vdtodate  
            vdtotime  
            status  
            ernam  
            erdat  
            aenam  
            aedat FROM j\_1ig\_ewaybill INTO TABLE gt\_ewaybill  
                              FOR ALL ENTRIES IN lt\_sd\_header  
                              WHERE bukrs =  lt\_sd\_header-bukrs  
                                AND doctyp = lt\_sd\_header-fkart  
                                AND docno  = lt\_sd\_header-vbeln.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_ewaybill DESCENDING BY egen\_dat egen\_time.  
      ENDIF.  
  
\* Document Flow Details  
      SELECT vbelv  posnv vbeln FROM vbfa  
               INTO TABLE gt\_vbfa  
               FOR ALL ENTRIES IN lt\_sd\_header  
               WHERE vbeln   = lt\_sd\_header-vbeln  
                 AND vbtyp\_v = gc\_vbtyp\_v.  
  
\* GSTIN deatils  
      SELECT  bukrs branch name adrnr gstin  
         FROM j\_1bbranch  
                INTO TABLE gt\_gstin FOR ALL ENTRIES IN lt\_sd\_header  
                WHERE bukrs = lt\_sd\_header-bukrs  
                  AND branch = lt\_sd\_header-bupla.  
  
\* Get Transporter details from Z-Table  
      SELECT \* FROM zteway\_transport INTO TABLE gt\_zteway\_transport  
                                    FOR ALL ENTRIES IN lt\_sd\_header  
                              WHERE bukrs =  lt\_sd\_header-bukrs  
                                AND doctyp = lt\_sd\_header-fkart  
                                AND docno  = lt\_sd\_header-vbeln.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_zteway\_transport BY bukrs doctyp docno.  
      ENDIF.  
      SELECT \* FROM likp INTO TABLE gt\_likp\_t  
        FOR ALL ENTRIES IN lt\_sd\_header  
        WHERE "bukrs = lt\_sd\_header-bukrs  
           vbeln = lt\_sd\_header-vgbel\_i.  
      IF gt\_likp\_t IS NOT INITIAL.  
        SELECT \* FROM dfkkbptaxnum  
        INTO TABLE gt\_dfkk  
         FOR ALL ENTRIES IN gt\_likp\_t  
          WHERE partner = gt\_likp\_t-zztrcd.  
        SELECT \* FROM zzroute  
          INTO TABLE gt\_zzroute  
          FOR ALL ENTRIES IN gt\_likp\_t  
          WHERE route = gt\_likp\_t-route.  
      ENDIF.  
\* Get e-invoice details from Z-Table  
      SELECT \* FROM zteinv\_details INTO TABLE gt\_zteinv\_details  
                                    FOR ALL ENTRIES IN lt\_sd\_header  
                              WHERE bukrs =  lt\_sd\_header-bukrs  
                                AND doctyp = lt\_sd\_header-fkart  
                                AND docno  = lt\_sd\_header-vbeln.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_zteinv\_details BY  bukrs docno doctyp.  
      ENDIF.  
  
  
  
\* Select partners  
      SELECT vbeln posnr parvw kunnr lifnr adrnr  
            FROM vbpa INTO TABLE gt\_vbpa  
            FOR ALL ENTRIES IN lt\_sd\_header  
            WHERE vbeln = lt\_sd\_header-vbeln.  
  
\*Buyer Details  
      IF gt\_vbpa IS NOT INITIAL.  
        SORT gt\_vbpa BY vbeln parvw.  
        SELECT  kunnr land1 name1 name2 ort01 pstlz  
                regio stras telf1 adrnr stcd3  
                 FROM kna1 INTO  TABLE gt\_kna1  
                 FOR ALL ENTRIES IN gt\_vbpa  
                 WHERE kunnr = gt\_vbpa-kunnr.  
        IF sy-subrc EQ 0.  
          SORT gt\_kna1 BY kunnr.  
          SELECT spras land1 bland bezei FROM t005u  
            INTO TABLE gt\_t005u  
            FOR ALL ENTRIES IN gt\_kna1  
            WHERE land1 EQ gt\_kna1-land1  
            AND bland EQ gt\_kna1-regio  
            AND spras EQ sy-langu.  
  
          SELECT addrnumber tel\_number  str\_suppl1 str\_suppl2 FROM adrc  
            INTO TABLE gt\_adrc  
            FOR ALL ENTRIES IN gt\_kna1  
            WHERE addrnumber = gt\_kna1-adrnr.  
  
          IF sy-subrc IS INITIAL.  
            SELECT addrnumber smtp\_addr   FROM adr6  
              INTO TABLE gt\_adr6  
              FOR ALL ENTRIES IN gt\_adrc  
              WHERE addrnumber = gt\_adrc-addrnumber.  
          ENDIF.  
        ENDIF.  
      ENDIF.  
  
      SELECT \* FROM j\_1ig\_invrefnum  
        INTO TABLE gt\_invrefnum  
        FOR ALL ENTRIES IN lt\_sd\_header  
        WHERE bukrs EQ lt\_sd\_header-bukrs  
      AND docno EQ lt\_sd\_header-vbeln  
      AND doc\_type EQ lt\_sd\_header-fkart.  
      IF sy-subrc EQ 0.  
        SORT  gt\_invrefnum DESCENDING.  
      ENDIF.  
  
  
  
      lt\_vbrk\_export = lt\_sd\_header.  
      DELETE lt\_vbrk\_export WHERE waerk = gc\_curr\_inr.  
  
      IF lt\_vbrk\_export IS NOT INITIAL.  
\*        SELECT bukrs  
\*                    belnr  
\*                    gjahr  
\*                    blart  
\*                    bldat  
\*                    budat  
\*                    tcode  
\*                    xblnr  
\*                    bktxt  
\*                    awtyp  
\*                    awkey FROM bkpf INTO TABLE gt\_bkpfm  
\*               FOR ALL ENTRIES IN lt\_vbrk\_export  
\*               WHERE bukrs = p\_ccode  
\*               AND  xblnr = lt\_vbrk\_export-xblnr.  
\*        IF gt\_bkpfm IS NOT INITIAL.  
\*          SELECT  bukrs  
\*                  belnr  
\*                  gjahr  
\*                  buzei  
\*                  buzid  
\*                  bschl  
\*                  koart  
\*                  shkzg  
\*                  dmbtr  
\*                  txgrp  
\*                  ktosl  
\*                  ebeln  
\*                  ebelp  
\*                  taxps  FROM bseg INTO TABLE gt\_bseg  
\*             FOR ALL ENTRIES IN gt\_bkpfm  
\*            WHERE bukrs = gt\_bkpfm-bukrs  
\*            AND belnr = gt\_bkpfm-belnr  
\*            AND gjahr = gt\_bkpfm-gjahr.  
\*        ENDIF.  
      ENDIF.  
    ENDIF.  
  
\* Get material details for HSN, and other fields  
    SELECT matnr werks steuc FROM marc INTO TABLE gt\_marc  
                       FOR ALL ENTRIES IN gt\_wb2\_v\_vbrk\_vbrp2  
                       WHERE matnr = gt\_wb2\_v\_vbrk\_vbrp2-matnr\_i  
                         AND werks = gt\_wb2\_v\_vbrk\_vbrp2-werks\_i.  
    IF sy-subrc IS INITIAL.  
      SORT gt\_marc BY matnr werks.  
    ENDIF.  
  
  
  
\* Order Details  
    IF gt\_vbfa IS NOT INITIAL.  
      SELECT  vbeln vbeln\_i posnr\_i audat bstnk bstdk ihrez bname  
               FROM wb2\_v\_vbak\_vbap2  
               INTO TABLE gt\_wb2\_v\_vbak\_vbap2  
               FOR ALL ENTRIES IN gt\_vbfa  
               WHERE vbeln = gt\_vbfa-vbelv.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_wb2\_v\_vbak\_vbap2 BY vbeln posnr\_i.  
      ENDIF.  
    ENDIF.  
  
\* Dispatch and  Ship Details  
    CLEAR:lt\_sd\_header.  
    lt\_sd\_header =  gt\_wb2\_v\_vbrk\_vbrp2.  
    SORT lt\_sd\_header BY vgbel\_i.  
    DELETE ADJACENT DUPLICATES FROM lt\_sd\_header COMPARING   vgbel\_i.  
    IF lt\_sd\_header IS NOT INITIAL.  
  
      SELECT vbeln kunnr FROM likp  
        INTO TABLE gt\_likp  
        FOR ALL ENTRIES IN lt\_sd\_header  
        WHERE vbeln EQ  lt\_sd\_header-vgbel\_i.  
  
      IF gt\_likp IS NOT INITIAL.  
  
        SELECT kunnr land1 name1 name2 ort01 pstlz regio stras telf1 adrnr stcd3  
                      FROM kna1 INTO TABLE gt\_kna1\_d  
                       FOR ALL ENTRIES IN gt\_likp  
                       WHERE kunnr = gt\_likp-kunnr.  
        IF sy-subrc EQ 0.  
\* Region text  
          SELECT  spras land1 land1 bezei FROM t005u  
            INTO TABLE gt\_t005u\_d  
            FOR ALL ENTRIES IN gt\_kna1\_d  
            WHERE land1 EQ gc\_land1\_in  
            AND bland EQ gt\_kna1\_d-regio  
            AND spras EQ sy-langu.  
  
\* Address details  
          SELECT addrnumber tel\_number FROM adrc  
            INTO TABLE gt\_adrc\_d  
            FOR ALL ENTRIES IN gt\_kna1\_d  
            WHERE addrnumber = gt\_kna1\_d-adrnr.  
        ENDIF.  
      ENDIF.  
    ENDIF.  
  
  
\*Seller Details  
    CLEAR lt\_sd\_header.  
    lt\_sd\_header = gt\_wb2\_v\_vbrk\_vbrp2.  
    SORT lt\_sd\_header BY werks\_i.  
  
    DELETE ADJACENT DUPLICATES FROM lt\_sd\_header COMPARING werks\_i.  
    IF lt\_sd\_header IS NOT INITIAL.  
  
      SELECT werks name1 name2 stras pstlz ort01 land1 regio  
          counc cityc adrnr spras  zone1 j\_1bbranch  
       FROM t001w  
     INTO TABLE gt\_t001w  
     FOR ALL ENTRIES IN lt\_sd\_header  
     WHERE werks EQ lt\_sd\_header-werks\_i.  
  
      IF sy-subrc EQ 0.  
        SORT gt\_t001w BY werks.  
  
\* Region text  
        SELECT spras land1 bland bezei FROM t005u  
          INTO TABLE gt\_t005u\_s  
          FOR ALL ENTRIES IN gt\_t001w  
          WHERE land1 EQ gt\_t001w-land1  
          AND bland EQ gt\_t001w-regio  
          AND spras EQ sy-langu.  
\* Address details  
        SELECT addrnumber tel\_number  FROM adrc  
          INTO  TABLE gt\_adrc\_s  
          FOR ALL ENTRIES IN gt\_t001w  
          WHERE addrnumber = gt\_t001w-adrnr.  
        IF sy-subrc EQ 0.  
          SELECT addrnumber smtp\_addr  
            FROM adr6  
            INTO TABLE gt\_adr6\_s  
            FOR ALL ENTRIES IN gt\_adrc\_s  
            WHERE addrnumber = gt\_adrc\_s-addrnumber.  
        ENDIF.  
      ENDIF.  
    ENDIF.  
  
  
\* Export region  
    IF gt\_export IS NOT INITIAL.  
      SELECT spras land1 bland bezei FROM t005u  
      APPENDING TABLE gt\_t005u  
      FOR ALL ENTRIES IN gt\_export  
      WHERE land1 EQ gt\_export-land1  
      AND bland EQ gt\_export-zport\_state  
      AND spras EQ sy-langu.  
    ENDIF.  
  
\* get unit rates and discounts  
    rs\_name-sign = gc\_sign.  
    rs\_name-option = gc\_equal.  
    rs\_name-low = gc\_var\_unit.  
    APPEND rs\_name TO rt\_name.  
    rs\_name-low = gc\_var\_disc.  
    APPEND rs\_name TO rt\_name.  
    SELECT  name type numb sign opti low high  
            FROM tvarvc INTO TABLE gt\_tvarvc  
            WHERE name IN rt\_name.  
  
  
  
    SORT gt\_wb2\_v\_vbrk\_vbrp2 BY vbeln.  
    lt\_vbrk\_vbrp = gt\_wb2\_v\_vbrk\_vbrp2.  
    DELETE ADJACENT DUPLICATES FROM lt\_vbrk\_vbrp COMPARING vbeln.  
  
    LOOP AT lt\_vbrk\_vbrp INTO lw\_wb2\_v\_vbrk\_vbrp2.  
  
      CLEAR:lw\_invrefnum.  
      READ TABLE gt\_invrefnum INTO lw\_invrefnum WITH KEY bukrs = lw\_wb2\_v\_vbrk\_vbrp2-bukrs  
                                                         docno = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                        doc\_type = lw\_wb2\_v\_vbrk\_vbrp2-fkart.  
      IF sy-subrc = 0.  
        gw\_final-ack\_no = lw\_invrefnum-ack\_no.  
  
        IF lw\_invrefnum-ack\_date IS NOT INITIAL.  
          lv\_internal\_date = lw\_invrefnum-ack\_date.  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lv\_internal\_date  
            IMPORTING  
              date\_external            = gw\_final-ack\_dt  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
        ENDIF.  
  
        gw\_final-qr\_code = lw\_invrefnum-signed\_qrcode.  
        gw\_final-irn = lw\_invrefnum-irn.  
        gw\_final-sign\_inv = lw\_invrefnum-signed\_inv.  
        gw\_final-ernam = lw\_invrefnum-ernam.  
  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_invrefnum-erdat  
          IMPORTING  
            date\_external            = gw\_final-erdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
        gw\_final-erzet = |{ lw\_invrefnum-erzet(2) }:{ lw\_invrefnum-erzet+2(2) }:{ lw\_invrefnum-erzet+4(2) }|."lw\_invrefnum-erzet.  
  
        IF lw\_invrefnum-cancel\_date IS NOT INITIAL.  
          lv\_internal\_date = lw\_invrefnum-cancel\_date.  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lv\_internal\_date  
            IMPORTING  
              date\_external            = gw\_final-canc\_dt  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
  
        ENDIF.  
        IF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_act.  
          gw\_final-status = 'Success'(060).  
          gw\_final-icon    = gc\_icon\_08.  
        ELSEIF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_err.  
          gw\_final-status = 'Error'(061).  
          gw\_final-icon    = gc\_icon\_0a.  
        ELSEIF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_cnl.  
          gw\_final-status = 'Cancelled'(062).  
          gw\_final-icon    = gc\_icon\_0w.  
        ENDIF.  
      ENDIF.  
  
      CLEAR ls\_zteinv\_details.  
      READ TABLE gt\_zteinv\_details INTO ls\_zteinv\_details WITH KEY  
                                                                     bukrs = lw\_wb2\_v\_vbrk\_vbrp2-bukrs  
                                                                     docno = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                                     doctyp = lw\_wb2\_v\_vbrk\_vbrp2-fkart  
                                                                     BINARY SEARCH.  
  
      IF sy-subrc = 0.  
        IF gw\_final-status = 'Error'(061).  
          gw\_final-einv\_error     =  ls\_zteinv\_details-einv\_error.  
        ELSE.  
          gw\_final-eway\_error     =  ' '.  
        ENDIF.  
      ENDIF.  
  
  
  
      CLEAR:lw\_ewaybill.  
      READ TABLE gt\_ewaybill INTO lw\_ewaybill WITH KEY bukrs = lw\_wb2\_v\_vbrk\_vbrp2-bukrs  
                                                                docno = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                                doctyp = lw\_wb2\_v\_vbrk\_vbrp2-fkart.  
      IF sy-subrc = 0.  
        gw\_final-eway\_num = lw\_ewaybill-ebillno.  
  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-egen\_dat  
          IMPORTING  
            date\_external            = gw\_final-eway\_date  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
        gw\_final-eway\_ernam = lw\_ewaybill-ernam.  
  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-erdat  
          IMPORTING  
            date\_external            = gw\_final-eway\_erdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
        IF lw\_ewaybill-status EQ  gc\_eway\_sts\_c.  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lw\_ewaybill-aedat  
            IMPORTING  
              date\_external            = gw\_final-eway\_canc\_dt  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
        ELSE.  
          gw\_final-eway\_canc\_dt = 0.  
        ENDIF.  
  
        gw\_final-eway\_erzet = |{ lw\_ewaybill-egen\_time(2) }:{ lw\_ewaybill-egen\_time+2(2) }:{ lw\_ewaybill-egen\_time+4(2) }|."lw\_ewaybill-egen\_time.  
  
\* Valid from  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-vdfmdate  
          IMPORTING  
            date\_external            = gw\_final-eway\_v\_from  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
\* Valid To  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-vdtodate  
          IMPORTING  
            date\_external            = gw\_final-eway\_v\_to  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
        gw\_final-eway\_v\_time = lw\_ewaybill-vdtotime.  
  
  
        IF lw\_ewaybill-status = gc\_eway\_sts\_a.  
          gw\_final-eway\_status = 'Success'(060).  
          gw\_final-eway\_icon    = gc\_icon\_08.  
        ELSEIF lw\_ewaybill-status = gc\_eway\_sts\_a.  
          gw\_final-eway\_status = 'Error'(061).  
          gw\_final-eway\_icon    = gc\_icon\_0a.  
        ELSEIF lw\_ewaybill-status = gc\_eway\_sts\_c.  
          gw\_final-eway\_status = 'Cancelled'(062).  
          gw\_final-eway\_icon    = gc\_icon\_0w.  
        ENDIF.  
      ENDIF.  
  
      READ TABLE gt\_likp\_t INTO gw\_likp\_t WITH KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vgbel\_i.  
      IF sy-subrc = 0.  
        gw\_final-t\_name  = gw\_likp\_t-zztrcd..  
        gw\_final-t\_doc\_no  = gw\_likp\_t-zzlrnum.  
        gw\_final-v\_number  = gw\_likp\_t-zzvehno.  
        gw\_final-t\_mode   = 'Road'.  
        DATA:lv\_date TYPE char10.  
        IF gw\_likp\_t-zzlrdat IS NOT INITIAL.  
  
          CONCATENATE gw\_likp\_t-zzlrdat+6(2) '/'  
                      gw\_likp\_t-zzlrdat+4(2) '/'  
                      gw\_likp\_t-zzlrdat+0(4)  
                      INTO gw\_final-t\_date.  
        ENDIF.  
        READ TABLE gt\_dfkk INTO gw\_dfkk WITH KEY partner = gw\_likp\_t-zztrcd.  
        IF sy-subrc = 0.  
          gw\_final-t\_id  = gw\_dfkk-taxnum."'05AAABB0639G1Z8'." gw\_dfkk-taxnum..  
        ENDIF.  
        READ TABLE gt\_zzroute INTO gw\_zzroute WITH KEY route = gw\_likp\_t-route.  
        IF sy-subrc = 0.  
          gw\_final-t\_distance = gw\_zzroute-kms\_ex."'0'." gw\_zzroute-kms\_ex.  
          CONDENSE gw\_final-t\_distance.  
        ENDIF.  
      ENDIF.  
  
      READ TABLE gt\_zteway\_transport INTO ls\_zteway\_transport WITH KEY  
                                                                bukrs = lw\_wb2\_v\_vbrk\_vbrp2-bukrs  
                                                                docno = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                                doctyp = lw\_wb2\_v\_vbrk\_vbrp2-fkart  
                                                                BINARY SEARCH.  
\*      IF sy-subrc = 0.  
\*        gw\_final-t\_id       =  ls\_zteway\_transport-t\_id.  
\*        gw\_final-t\_name     =  ls\_zteway\_transport-t\_name.  
\*        gw\_final-t\_doc\_no   =  ls\_zteway\_transport-t\_doc\_no.  
  
\*  
\*        IF ls\_zteway\_transport-t\_date IS NOT INITIAL.  
\*          CONCATENATE ls\_zteway\_transport-t\_date+6(2) '/'  
\*                      ls\_zteway\_transport-t\_date+4(2) '/'  
\*                      ls\_zteway\_transport-t\_date+0(4)  
\*                      INTO gw\_final-t\_date.  
\*        ENDIF.  
  
\*        gw\_final-t\_mode     =  ls\_zteway\_transport-t\_mode.  
\*                                                gw\_final-t\_distance =  ls\_zteway\_transport-t\_distance.  
\*        gw\_final-v\_number   =  ls\_zteway\_transport-v\_number.  
      gw\_final-v\_type     =  ls\_zteway\_transport-v\_type.  
      IF gw\_final-v\_type  IS INITIAL.  
        gw\_final-v\_type     = gc\_v\_type\_1..  
      ENDIF.  
      gw\_final-eway\_print     =  ls\_zteway\_transport-eway\_print.  
      IF gw\_final-eway\_status = 'Error'(061).  
        gw\_final-eway\_error     =  ls\_zteway\_transport-eway\_error.  
      ELSE.  
        gw\_final-eway\_error     =  ' '.  
      ENDIF.  
      gw\_final-t\_r\_distance           = ls\_zteway\_transport-t\_r\_distance.  
      gw\_final-t\_ext\_valid\_reason     = ls\_zteway\_transport-t\_ext\_valid\_reason.  
      gw\_final-t\_ext\_valid\_remarks    = ls\_zteway\_transport-t\_ext\_valid\_remarks.  
      gw\_final-t\_from\_pin             = ls\_zteway\_transport-t\_from\_pin.  
      gw\_final-t\_consignment\_status   = ls\_zteway\_transport-t\_consignment\_status.  
      gw\_final-t\_transit\_type         = ls\_zteway\_transport-t\_transit\_type.  
      gw\_final-t\_address1             = ls\_zteway\_transport-t\_address1.  
      gw\_final-t\_address2             = ls\_zteway\_transport-t\_address2.  
      gw\_final-t\_address3             = ls\_zteway\_transport-t\_address3.  
  
      gw\_final-v\_reason\_code        = ls\_zteway\_transport-v\_reason\_code.  
      gw\_final-v\_reason             = ls\_zteway\_transport-v\_reason.  
      gw\_final-c\_reason\_code        = ls\_zteway\_transport-c\_reason\_code.  
      gw\_final-c\_reason             = ls\_zteway\_transport-c\_reason.  
  
      READ TABLE gt\_vbfa INTO DATA(gw\_vbfa) WITH KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
      IF sy-subrc = 0.  
        SELECT SINGLE port\_code FROM vbak INTO gw\_final-zport\_code WHERE vbeln =  gw\_vbfa-vbelv."lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
        "gw\_final-zport\_code             = ls\_zteway\_transport-zport\_code.  
      ENDIF.  
\*      ELSE.  
\* Customer specific logic  
\*      ENDIF.  
  
  
      gw\_final-vbeln      =        lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
      gw\_final-gjahr      =        lw\_wb2\_v\_vbrk\_vbrp2-gjahr.  
      gw\_final-fkart      =        lw\_wb2\_v\_vbrk\_vbrp2-fkart.  
      gw\_final-fktyp      =        lw\_wb2\_v\_vbrk\_vbrp2-fktyp.  
      gw\_final-vbtyp      =        lw\_wb2\_v\_vbrk\_vbrp2-vbtyp.  
      gw\_final-vkorg      =        lw\_wb2\_v\_vbrk\_vbrp2-vkorg.  
      gw\_final-vtweg     =        lw\_wb2\_v\_vbrk\_vbrp2-vtweg.  
      IF lw\_wb2\_v\_vbrk\_vbrp2-fkdat IS NOT INITIAL.  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_wb2\_v\_vbrk\_vbrp2-fkdat  
          IMPORTING  
            date\_external            = gw\_final-fkdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
        gw\_final-fkdat\_db = lw\_wb2\_v\_vbrk\_vbrp2-fkdat.  
      ENDIF.  
  
      gw\_final-kunrg     =        lw\_wb2\_v\_vbrk\_vbrp2-kunrg.  
      gw\_final-werks     =        lw\_wb2\_v\_vbrk\_vbrp2-werks\_i.  
      gw\_final-prctr     =        lw\_wb2\_v\_vbrk\_vbrp2-prctr\_i.  
      gw\_final-doc\_ernam =        lw\_wb2\_v\_vbrk\_vbrp2-ernam.  
      gw\_final-bupla     =        lw\_wb2\_v\_vbrk\_vbrp2-bupla.  
      gw\_final-odnno     =        lw\_wb2\_v\_vbrk\_vbrp2-xblnr.  
      gw\_final-bukrs     =        lw\_wb2\_v\_vbrk\_vbrp2-bukrs.  
  
\* Get Fiscl year  
      CALL FUNCTION 'GM\_GET\_FISCAL\_YEAR'  
        EXPORTING  
          i\_date                     = lw\_wb2\_v\_vbrk\_vbrp2-fkdat  
          i\_fyv                      = gc\_periv\_v3  
        IMPORTING  
          e\_fy                       = lv\_gjahr  
        EXCEPTIONS  
          fiscal\_year\_does\_not\_exist = 1  
          not\_defined\_for\_date       = 2  
          OTHERS                     = 3.  
      IF sy-subrc <> 0.  
\* Implement suitable error handling here  
      ELSE.  
        gw\_final-gjahr = lv\_gjahr.  
      ENDIF.  
  
  
\* Ship to details  
      CLEAR:lw\_vbpa.  
      READ TABLE gt\_vbpa INTO lw\_vbpa WITH  KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                parvw = gc\_parvw\_we  
                                                BINARY SEARCH.  
      IF sy-subrc EQ 0.  
        CLEAR:lw\_kna1\_d.  
        READ TABLE gt\_kna1 INTO lw\_kna1\_d WITH  KEY kunnr = lw\_vbpa-kunnr BINARY SEARCH.  
        IF sy-subrc EQ 0.  
          gw\_final-ship\_to = lw\_kna1\_d-kunnr.  
          gw\_final-ship\_to\_name = lw\_kna1\_d-name1.  
        ENDIF.  
      ENDIF.  
  
  
      READ TABLE gt\_gstin INTO lw\_gstin WITH KEY bukrs = gw\_final-bukrs  
                                                 branch = gw\_final-bupla.  
      IF sy-subrc IS INITIAL.  
        gw\_final-sup\_gstin = lw\_gstin-gstin.  
      ENDIF.  
  
      READ TABLE gt\_kna1 INTO lw\_kna1 WITH KEY kunnr = gw\_final-kunrg BINARY SEARCH.  
      IF sy-subrc IS INITIAL.  
        gw\_final-kunnr     =        lw\_kna1-kunnr.  
        gw\_final-name1     =        lw\_kna1-name1.  
        gw\_final-ship\_gstin = lw\_kna1-stcd3.  
      ENDIF.  
  
      READ TABLE gt\_doctyp INTO lw\_doctyp WITH KEY bukrs = p\_ccode  
                                                   zmodule = gc\_sd  
                                                   fkart   = gw\_final-fkart.  
      IF sy-subrc IS INITIAL.  
        gw\_final-einv = lw\_doctyp-einv.  
        gw\_final-eway = lw\_doctyp-eway.  
      ENDIF.  
  
      gw\_final-mod = gc\_sd.  
  
      READ TABLE gt\_wb2\_v\_vbrk\_vbrp2 INTO lw\_vbrk\_vbrp2 WITH KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
      IF sy-subrc = 0.  
        lv\_index = sy-tabix.  
        LOOP AT gt\_wb2\_v\_vbrk\_vbrp2 INTO lw\_vbrk\_vbrp2 FROM lv\_index.  
          IF lw\_vbrk\_vbrp2-vbeln NE  lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
            EXIT.  
          ENDIF.  
  
  
\* check for export currency  
          CLEAR lv\_exp\_curr.  
          IF lw\_vbrk\_vbrp2-waerk NE  gc\_curr\_inr .  
            lv\_exp\_curr = abap\_true.  
          ENDIF.  
  
          gw\_final-taxamt    =  gw\_final-taxamt + lw\_vbrk\_vbrp2-netwr\_i.  
          gw\_final-tax       =  gw\_final-tax    + lw\_vbrk\_vbrp2-mwsbp\_i.  
          "IGST Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_vbrk\_vbrp2-knumv  
                                                   kposn = lw\_vbrk\_vbrp2-posnr\_i  
                                                   kschl = gc\_kschl\_joig  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-igst\_amt  =  gw\_final-igst\_amt + lw\_konv-kwert.  
          ENDIF.  
          "CGST Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_vbrk\_vbrp2-knumv  
                                                   kposn = lw\_vbrk\_vbrp2-posnr\_i  
                                                   kschl = gc\_kschl\_jocg  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-cgst\_amt =  gw\_final-cgst\_amt + lw\_konv-kwert.  
          ENDIF.  
          "SGST Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_vbrk\_vbrp2-knumv  
                                                   kposn = lw\_vbrk\_vbrp2-posnr\_i  
                                                   kschl = gc\_kschl\_josg  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-sgst\_amt =  gw\_final-sgst\_amt + lw\_konv-kwert.  
          ENDIF.  
          "CESS Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_vbrk\_vbrp2-knumv  
                                                   kposn = lw\_vbrk\_vbrp2-posnr\_i  
                                                   kschl = gc\_kschl\_cess  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-cess\_amt =  gw\_final-cess\_amt + lw\_konv-kwert.  
          ENDIF.  
  
        ENDLOOP.  
  
        gw\_final-invval =  gw\_final-taxamt + gw\_final-tax .  
  
      ENDIF.  
  
  
      IF lv\_exp\_curr IS NOT INITIAL.  
        PERFORM convert\_currency USING gw\_final-invval lw\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING gw\_final-invval.  
        PERFORM convert\_currency USING gw\_final-taxamt lw\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING gw\_final-taxamt.  
        PERFORM convert\_currency USING gw\_final-igst\_amt lw\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING gw\_final-igst\_amt.  
        PERFORM convert\_currency USING gw\_final-cgst\_amt lw\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING gw\_final-cgst\_amt.  
        PERFORM convert\_currency USING gw\_final-sgst\_amt lw\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING gw\_final-sgst\_amt.  
        PERFORM convert\_currency USING gw\_final-cess\_amt lw\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING gw\_final-cess\_amt.  
      ENDIF.  
  
      APPEND gw\_final TO gt\_final.  
      CLEAR : gw\_final.  
    ENDLOOP.  
  
  ENDIF.  
  
  
  
ENDFORM.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  PROCESS\_SD\_DATA\_DELIVERY  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM process\_sd\_data\_delivery .  
  
  
  DATA: lw\_invrefnum        TYPE j\_1ig\_invrefnum,  
        lw\_wb2\_v\_likp\_lips2 TYPE ty\_wb2\_v\_likp\_lips2,  
        lw\_kna1             LIKE LINE OF gt\_kna1,  
        lw\_gstin            LIKE LINE OF gt\_gstin.  
  DATA: lv\_internal\_date TYPE sy-datum,  
        lw\_konv          TYPE ty\_konv,  
        lv\_gjahr(4)      TYPE c,  
        lw\_likp          TYPE ty\_likp,  
        lw\_lfa1          TYPE lfa1,  
        lw\_doctyp        TYPE ty\_doctyp.  
  DATA: lw\_ewaybill         TYPE ty\_ewaybill.  
  DATA: ls\_zteway\_transport TYPE zteway\_transport.  
  DATA: ls\_zteinv\_details   TYPE zteinv\_details.  
  DATA:lt\_likp\_lips2      TYPE TABLE OF ty\_wb2\_v\_likp\_lips2,  
       lw\_likp\_lips2      TYPE  ty\_wb2\_v\_likp\_lips2,  
       lt\_delivery\_header TYPE TABLE OF ty\_wb2\_v\_likp\_lips2,  
       lv\_index           TYPE sy-tabix.  
  
  
\*& Get deliveries for E-Way Bill  
  IF s\_prctr IS NOT INITIAL.  
    SELECT vbeln vbeln\_i posnr\_i ernam vkorg lfart lfdat vbtyp kunnr kunag  
           fkdat knumv waerk lifnr bldat netwr werks xblnr  matnr\_i matkl\_i  
           werks\_i lgort\_i charg\_i lfimg\_i meins\_i vrkme\_i ntgew\_i brgew\_i  
           arktx\_i vbelv\_i posnv\_i vgbel\_i vgpos\_i gsber\_i  prctr\_i  
           netpr\_i netwr\_i lfgja\_i  
        FROM wb2\_v\_likp\_lips2  
        INTO TABLE gt\_wb2\_v\_likp\_lips2  
        FOR ALL ENTRIES IN gt\_user\_config  
        WHERE lfart = gt\_user\_config-fkart  
        AND vbeln IN s\_doc  
        AND fkdat IN s\_date  
        AND werks\_i = gt\_user\_config-werks  
        AND prctr\_i IN s\_prctr.  
  
  ELSE.  
    SELECT vbeln vbeln\_i posnr\_i ernam vkorg lfart lfdat vbtyp kunnr kunag  
           fkdat knumv waerk lifnr bldat netwr werks xblnr  matnr\_i matkl\_i  
           werks\_i lgort\_i charg\_i lfimg\_i meins\_i vrkme\_i ntgew\_i brgew\_i  
           arktx\_i vbelv\_i posnv\_i vgbel\_i vgpos\_i gsber\_i  prctr\_i  
           netpr\_i netwr\_i lfgja\_i  
          FROM wb2\_v\_likp\_lips  
             INTO TABLE gt\_wb2\_v\_likp\_lips2  
             FOR ALL ENTRIES IN gt\_user\_config  
             WHERE lfart = gt\_user\_config-fkart  
             AND vbeln IN s\_doc  
             AND fkdat IN s\_date  
             AND werks\_i = gt\_user\_config-werks.  
  ENDIF.  
  
  
  
  IF gt\_wb2\_v\_likp\_lips2 IS  NOT INITIAL.  
  
  
    lt\_delivery\_header = gt\_wb2\_v\_likp\_lips2.  
    SORT lt\_delivery\_header BY  lfart vbeln.  
    DELETE ADJACENT DUPLICATES FROM lt\_delivery\_header COMPARING lfart vbeln.  
    IF lt\_delivery\_header IS NOT INITIAL .  
  
\* Eway bill  
      SELECT bukrs doctyp docno gjahr ebillno egen\_dat egen\_time vdfmdate vdtodate  
             vdtotime status ernam erdat aenam aedat  
             FROM j\_1ig\_ewaybill APPENDING TABLE gt\_ewaybill  
             FOR ALL ENTRIES IN lt\_delivery\_header  
              WHERE bukrs = p\_ccode  
              AND doctyp = lt\_delivery\_header-lfart  
              AND docno  = lt\_delivery\_header-vbeln.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_ewaybill DESCENDING BY egen\_dat egen\_time.  
      ENDIF.  
  
\* Get Transporter details from Z-Table  
      SELECT \* FROM zteway\_transport INTO TABLE gt\_zteway\_transport  
                                    FOR ALL ENTRIES IN lt\_delivery\_header  
                              WHERE bukrs =  p\_ccode  
                               AND doctyp = lt\_delivery\_header-lfart  
                                AND docno  = lt\_delivery\_header-vbeln.  
      IF sy-subrc IS INITIAL.  
        SORT gt\_zteway\_transport BY bukrs doctyp docno.  
      ENDIF.  
  
\* Select partners  
      SELECT vbeln posnr parvw kunnr lifnr  
          adrnr   FROM vbpa APPENDING TABLE gt\_vbpa FOR ALL ENTRIES IN lt\_delivery\_header  
                                  WHERE vbeln = lt\_delivery\_header-vbeln.  
  
\*Buyer Details  
      IF sy-subrc IS INITIAL.  
        SORT gt\_vbpa BY vbeln parvw.  
        SELECT kunnr land1 name1 name2 ort01 pstlz regio  
             stras telf1 adrnr stcd3  
           FROM kna1  
                 APPENDING TABLE gt\_kna1  
                 FOR ALL ENTRIES IN gt\_vbpa  
                 WHERE kunnr = gt\_vbpa-kunnr.  
        IF sy-subrc EQ 0.  
          SORT gt\_kna1 BY kunnr.  
  
          SELECT spras land1 bland bezei  FROM t005u  
            APPENDING TABLE gt\_t005u  
            FOR ALL ENTRIES IN gt\_kna1  
            WHERE land1 EQ gt\_kna1-land1  
            AND bland EQ gt\_kna1-regio  
            AND spras EQ sy-langu.  
  
          SELECT addrnumber tel\_number FROM adrc  
            APPENDING TABLE gt\_adrc  
            FOR ALL ENTRIES IN gt\_kna1  
            WHERE addrnumber = gt\_kna1-adrnr.  
  
          IF sy-subrc IS INITIAL.  
            SELECT addrnumber smtp\_addr   FROM adr6  
              APPENDING  TABLE gt\_adr6  
              FOR ALL ENTRIES IN gt\_adrc  
              WHERE addrnumber = gt\_adrc-addrnumber.  
          ENDIF.  
        ENDIF.  
      ENDIF.  
  
      SELECT \* FROM j\_1ig\_invrefnum  
          APPENDING TABLE gt\_invrefnum  
          FOR ALL ENTRIES IN lt\_delivery\_header  
          WHERE bukrs EQ p\_ccode  
        AND docno EQ lt\_delivery\_header-vbeln  
        AND doc\_type EQ lt\_delivery\_header-lfart.  
      IF sy-subrc EQ 0.  
        SORT  gt\_invrefnum BY bukrs docno doc\_year doc\_type.  
      ENDIF.  
    ENDIF.  
  
  
\* Get material details for HSN, and other fields  
  
    SELECT matnr werks steuc FROM marc APPENDING TABLE gt\_marc  
                       FOR ALL ENTRIES IN gt\_wb2\_v\_likp\_lips2  
                       WHERE matnr = gt\_wb2\_v\_likp\_lips2-matnr\_i  
                         AND werks = gt\_wb2\_v\_likp\_lips2-werks\_i.  
    IF sy-subrc IS INITIAL.  
      SORT gt\_marc BY matnr werks.  
    ENDIF.  
  
  
\* GSTIN deatils  
    SELECT bukrs branch name adrnr gstin  
        FROM j\_1bbranch  
              APPENDING TABLE gt\_gstin  
              WHERE bukrs = p\_ccode.  
  
  
    CLEAR:lt\_delivery\_header.  
    lt\_delivery\_header = gt\_wb2\_v\_likp\_lips2.  
    SORT lt\_delivery\_header BY kunnr.  
    DELETE ADJACENT DUPLICATES FROM lt\_delivery\_header COMPARING kunnr.  
  
    IF lt\_delivery\_header IS NOT INITIAL.  
      SELECT kunnr land1 name1 name2 ort01 pstlz regio  
             stras telf1 adrnr stcd3  
          FROM kna1  
                     APPENDING TABLE gt\_kna1\_d  
                     FOR ALL ENTRIES IN gt\_wb2\_v\_likp\_lips2  
                     WHERE kunnr = gt\_wb2\_v\_likp\_lips2-kunnr.  
      IF sy-subrc EQ 0.  
  
        SELECT   spras  
          land1  
          bland  
          bezei   FROM t005u  
          APPENDING TABLE gt\_t005u\_d  
          FOR ALL ENTRIES IN gt\_kna1\_d  
          WHERE land1 EQ gc\_land1\_in  
          AND bland EQ gt\_kna1\_d-regio  
          AND spras EQ sy-langu.  
  
        SELECT addrnumber tel\_number  FROM adrc  
          APPENDING TABLE gt\_adrc\_d  
          FOR ALL ENTRIES IN gt\_kna1\_d  
          WHERE addrnumber = gt\_kna1\_d-adrnr.  
  
      ENDIF.  
    ENDIF.  
  
\*Seller Details  
  
    CLEAR:lt\_delivery\_header.  
    lt\_delivery\_header = gt\_wb2\_v\_likp\_lips2.  
    SORT lt\_delivery\_header BY werks\_i.  
    DELETE ADJACENT DUPLICATES FROM lt\_delivery\_header COMPARING werks\_i.  
  
    IF lt\_delivery\_header IS NOT INITIAL.  
      SELECT werks name1 name2 stras pstlz ort01 land1  
             regio counc cityc adrnr spras zone1  
           j\_1bbranch FROM t001w  
        APPENDING TABLE gt\_t001w  
        FOR ALL ENTRIES IN gt\_wb2\_v\_likp\_lips2  
        WHERE werks EQ gt\_wb2\_v\_likp\_lips2-werks\_i.  
      IF sy-subrc EQ 0.  
        SORT gt\_t001w BY werks.  
        SELECT spras land1 bland bezei FROM t005u  
          APPENDING TABLE gt\_t005u\_s  
          FOR ALL ENTRIES IN gt\_t001w  
          WHERE land1 EQ gt\_t001w-land1  
          AND bland EQ gt\_t001w-regio  
          AND spras EQ sy-langu.  
  
        SELECT addrnumber tel\_number  FROM adrc  
          APPENDING TABLE gt\_adrc\_s  
          FOR ALL ENTRIES IN gt\_t001w  
          WHERE addrnumber = gt\_t001w-adrnr.  
        IF sy-subrc EQ 0.  
          SELECT addrnumber smtp\_addr   FROM adr6  
            APPENDING TABLE  gt\_adr6\_s  
            FOR ALL ENTRIES IN gt\_adrc\_s  
            WHERE addrnumber = gt\_adrc\_s-addrnumber.  
        ENDIF.  
      ENDIF.  
    ENDIF.  
  
  
  
  
    SORT gt\_wb2\_v\_likp\_lips2 BY vbeln.  
    lt\_likp\_lips2 = gt\_wb2\_v\_likp\_lips2.  
    DELETE ADJACENT DUPLICATES FROM lt\_likp\_lips2 COMPARING vbeln.  
  
    LOOP AT gt\_wb2\_v\_likp\_lips2 INTO lw\_wb2\_v\_likp\_lips2.  
  
      CLEAR:lw\_invrefnum.  
      READ TABLE gt\_invrefnum INTO lw\_invrefnum WITH KEY bukrs = p\_ccode  
                                                                docno = lw\_wb2\_v\_likp\_lips2-vbeln  
                                                                doc\_type = lw\_wb2\_v\_likp\_lips2-lfart.  
      IF sy-subrc = 0.  
        gw\_final-ack\_no = lw\_invrefnum-ack\_no.  
  
        IF lw\_invrefnum-ack\_date IS NOT INITIAL.  
          lv\_internal\_date = lw\_invrefnum-ack\_date.  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lv\_internal\_date  
            IMPORTING  
              date\_external            = gw\_final-ack\_dt  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
        ENDIF.  
  
        gw\_final-qr\_code = lw\_invrefnum-signed\_qrcode.  
        gw\_final-irn = lw\_invrefnum-irn.  
        gw\_final-sign\_inv = lw\_invrefnum-signed\_inv.  
        gw\_final-ernam = lw\_invrefnum-ernam.  
  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_invrefnum-erdat  
          IMPORTING  
            date\_external            = gw\_final-erdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
        gw\_final-erzet = lw\_invrefnum-erzet.  
  
        IF lw\_invrefnum-cancel\_date IS NOT INITIAL.  
          lv\_internal\_date = lw\_invrefnum-cancel\_date.  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lv\_internal\_date  
            IMPORTING  
              date\_external            = gw\_final-canc\_dt  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
  
        ENDIF.  
        IF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_act.  
          gw\_final-status = 'Success'(060).  
          gw\_final-icon    = gc\_icon\_08.  
        ELSEIF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_err.  
          gw\_final-status = 'Error'(061).  
          gw\_final-icon    = gc\_icon\_0a.  
        ELSEIF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_cnl.  
          gw\_final-status = 'Cancelled'(062).  
          gw\_final-icon   = gc\_icon\_0w.  
        ENDIF.  
      ENDIF.  
  
  
      CLEAR  ls\_zteinv\_details.  
      READ TABLE gt\_zteinv\_details INTO ls\_zteinv\_details WITH KEY  
                                                                bukrs = p\_ccode  
                                                                docno = lw\_wb2\_v\_likp\_lips2-vbeln  
                                                                doctyp = lw\_wb2\_v\_likp\_lips2-lfart.  
      IF sy-subrc = 0.  
        IF gw\_final-status = 'Error'(061).  
          gw\_final-einv\_error     =  ls\_zteinv\_details-einv\_error.  
        ELSE.  
          gw\_final-einv\_error     =  ' '.  
        ENDIF.  
      ENDIF.  
  
  
  
      CLEAR:lw\_ewaybill.  
      READ TABLE gt\_ewaybill INTO lw\_ewaybill WITH KEY bukrs = p\_ccode  
                                                                docno = lw\_wb2\_v\_likp\_lips2-vbeln  
                                                                doctyp = lw\_wb2\_v\_likp\_lips2-lfart.  
      IF sy-subrc = 0.  
        gw\_final-eway\_num = lw\_ewaybill-ebillno.  
  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-egen\_dat  
          IMPORTING  
            date\_external            = gw\_final-eway\_date  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
        gw\_final-eway\_ernam = lw\_ewaybill-ernam.  
  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-erdat  
          IMPORTING  
            date\_external            = gw\_final-eway\_erdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
        IF lw\_ewaybill-status EQ  gc\_eway\_sts\_c.  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lw\_ewaybill-aedat  
            IMPORTING  
              date\_external            = gw\_final-eway\_canc\_dt  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
        ELSE.  
          gw\_final-eway\_canc\_dt = 0.  
        ENDIF.  
  
        gw\_final-eway\_erzet = |{ lw\_ewaybill-egen\_time(2) }:{ lw\_ewaybill-egen\_time+2(2) }:{ lw\_ewaybill-egen\_time+4(2) }|." lw\_ewaybill-egen\_time.  
  
\* Valid from  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-vdfmdate  
          IMPORTING  
            date\_external            = gw\_final-eway\_v\_from  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
  
\* Valid To  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_ewaybill-vdtodate  
          IMPORTING  
            date\_external            = gw\_final-eway\_v\_to  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
        gw\_final-eway\_v\_time = lw\_ewaybill-vdtotime.  
  
  
        IF lw\_ewaybill-status = gc\_eway\_sts\_a.  
          gw\_final-eway\_status = 'Success'(060).  
          gw\_final-eway\_icon    = gc\_icon\_08.  
        ELSEIF lw\_ewaybill-status = gc\_eway\_sts\_e.  
          gw\_final-eway\_status = 'Error'(061).  
          gw\_final-eway\_icon    = gc\_icon\_0a.  
        ELSEIF lw\_ewaybill-status = gc\_eway\_sts\_c.  
          gw\_final-eway\_status = 'Cancelled'(062).  
          gw\_final-eway\_icon    = gc\_icon\_0w.  
        ENDIF.  
      ENDIF.  
      READ TABLE gt\_likp\_t INTO gw\_likp\_t WITH KEY vbeln = lw\_wb2\_v\_likp\_lips2-vbeln.  
      IF sy-subrc = 0.  
        gw\_final-t\_name  = gw\_likp\_t-zztrcd..  
        gw\_final-t\_doc\_no  = gw\_likp\_t-zzlrnum.  
        gw\_final-v\_number  = gw\_likp\_t-zzvehno.  
        gw\_final-t\_mode   = 'Road'.  
        DATA:lv\_date TYPE char10.  
        IF gw\_likp\_t-zzlrdat IS NOT INITIAL.  
  
          CONCATENATE gw\_likp\_t-zzlrdat+6(2) '/'  
                      gw\_likp\_t-zzlrdat+4(2) '/'  
                      gw\_likp\_t-zzlrdat+0(4)  
                      INTO gw\_final-t\_date.  
        ENDIF.  
        READ TABLE gt\_dfkk INTO gw\_dfkk WITH KEY partner = gw\_likp\_t-zztrcd.  
        IF sy-subrc = 0.  
          gw\_final-t\_id  =  gw\_dfkk-taxnum."'05AAABB0639G1Z8'."gw\_dfkk-taxnum.  
        ENDIF.  
        READ TABLE gt\_zzroute INTO gw\_zzroute WITH KEY route = gw\_likp\_t-route.  
        IF sy-subrc = 0.  
          gw\_final-t\_distance =  gw\_zzroute-kms\_ex."'0'."gw\_zzroute-kms\_ex.  
          CONDENSE gw\_final-t\_distance.  
        ENDIF.  
      ENDIF.  
      READ TABLE gt\_zteway\_transport INTO ls\_zteway\_transport WITH KEY  
                                                                bukrs = p\_ccode  
                                                                docno = lw\_wb2\_v\_likp\_lips2-vbeln  
                                                                doctyp = lw\_wb2\_v\_likp\_lips2-lfart  
                                                                BINARY SEARCH.  
      IF sy-subrc = 0.  
\*        gw\_final-t\_id       =  ls\_zteway\_transport-t\_id.  
\*        gw\_final-t\_name     =  ls\_zteway\_transport-t\_name.  
\*        gw\_final-t\_doc\_no   =  ls\_zteway\_transport-t\_doc\_no.  
\*  
\*        IF ls\_zteway\_transport-t\_date IS NOT INITIAL.  
\*          CONCATENATE ls\_zteway\_transport-t\_date+6(2) '/'  
\*                      ls\_zteway\_transport-t\_date+4(2) '/'  
\*                      ls\_zteway\_transport-t\_date+0(4)  
\*                      INTO gw\_final-t\_date.  
\*        ENDIF.  
  
\*        gw\_final-t\_mode     =  ls\_zteway\_transport-t\_mode.  
\*                                              gw\_final-t\_distance =  ls\_zteway\_transport-t\_distance.  
\*        gw\_final-v\_number   =  ls\_zteway\_transport-v\_number.  
        gw\_final-v\_type     =  ls\_zteway\_transport-v\_type.  
  
        gw\_final-t\_r\_distance           = ls\_zteway\_transport-t\_r\_distance.  
        gw\_final-t\_ext\_valid\_reason     = ls\_zteway\_transport-t\_ext\_valid\_reason.  
        gw\_final-t\_ext\_valid\_remarks    = ls\_zteway\_transport-t\_ext\_valid\_remarks.  
        gw\_final-t\_from\_pin             = ls\_zteway\_transport-t\_from\_pin.  
        gw\_final-t\_consignment\_status   = ls\_zteway\_transport-t\_consignment\_status.  
        gw\_final-t\_transit\_type         = ls\_zteway\_transport-t\_transit\_type.  
        gw\_final-t\_address1             = ls\_zteway\_transport-t\_address1.  
        gw\_final-t\_address2             = ls\_zteway\_transport-t\_address2.  
        gw\_final-t\_address3             = ls\_zteway\_transport-t\_address3.  
  
        gw\_final-v\_reason\_code        = ls\_zteway\_transport-v\_reason\_code.  
        gw\_final-v\_reason             = ls\_zteway\_transport-v\_reason.  
        gw\_final-c\_reason\_code        = ls\_zteway\_transport-c\_reason\_code.  
        gw\_final-c\_reason             = ls\_zteway\_transport-c\_reason.  
  
        gw\_final-zport\_code               = ls\_zteway\_transport-zport\_code.  
  
      ELSE.  
  
\* Customer specific logic  
  
      ENDIF.  
  
      gw\_final-vbeln      =        lw\_wb2\_v\_likp\_lips2-vbeln.  
      gw\_final-fkart      =        lw\_wb2\_v\_likp\_lips2-lfart.  
      gw\_final-vbtyp      =        lw\_wb2\_v\_likp\_lips2-vbtyp.  
      gw\_final-vkorg      =        lw\_wb2\_v\_likp\_lips2-vkorg.  
  
      IF lw\_wb2\_v\_likp\_lips2-fkdat IS NOT INITIAL.  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = lw\_wb2\_v\_likp\_lips2-fkdat  
          IMPORTING  
            date\_external            = gw\_final-fkdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
        gw\_final-fkdat\_db = lw\_wb2\_v\_likp\_lips2-fkdat.  
      ENDIF.  
  
      gw\_final-kunrg     =        lw\_wb2\_v\_likp\_lips2-kunnr.  
      gw\_final-werks     =        lw\_wb2\_v\_likp\_lips2-werks\_i.  
      gw\_final-prctr     =        lw\_wb2\_v\_likp\_lips2-prctr\_i.  
      gw\_final-doc\_ernam =        lw\_wb2\_v\_likp\_lips2-ernam.  
      gw\_final-odnno     =        lw\_wb2\_v\_likp\_lips2-xblnr.  
      gw\_final-bukrs     =        p\_ccode.  
  
\* Get Fiscal year  
      CALL FUNCTION 'GM\_GET\_FISCAL\_YEAR'  
        EXPORTING  
          i\_date                     = lw\_wb2\_v\_likp\_lips2-fkdat  
          i\_fyv                      = gc\_periv\_v3  
        IMPORTING  
          e\_fy                       = lv\_gjahr  
        EXCEPTIONS  
          fiscal\_year\_does\_not\_exist = 1  
          not\_defined\_for\_date       = 2  
          OTHERS                     = 3.  
      IF sy-subrc <> 0.  
        MESSAGE 'Fiscal Year not found'(m03) TYPE 'I'.  
      ELSE.  
        gw\_final-gjahr = lv\_gjahr.  
      ENDIF.  
  
  
  
      READ TABLE gt\_gstin INTO lw\_gstin WITH KEY bukrs = gw\_final-bukrs.  
  
      IF sy-subrc IS INITIAL.  
        gw\_final-sup\_gstin = lw\_gstin-gstin.  
      ENDIF.  
  
  
  
      READ TABLE gt\_kna1 INTO lw\_kna1 WITH KEY kunnr = gw\_final-kunrg BINARY SEARCH.  
      IF sy-subrc IS INITIAL.  
        gw\_final-kunnr     =        lw\_kna1-kunnr.  
        gw\_final-name1     =        lw\_kna1-name1.  
        gw\_final-ship\_gstin = lw\_kna1-stcd3.  
      ENDIF.  
  
      READ TABLE gt\_doctyp INTO lw\_doctyp WITH KEY bukrs = p\_ccode  
                                                   zmodule = gc\_sd  
                                                   fkart   = gw\_final-fkart.  
      IF sy-subrc IS INITIAL.  
        gw\_final-einv = lw\_doctyp-einv.  
        gw\_final-eway = lw\_doctyp-eway.  
      ENDIF.  
      gw\_final-mod = gc\_sd.  
  
  
      READ TABLE gt\_wb2\_v\_likp\_lips2 INTO lw\_likp\_lips2 WITH KEY vbeln = lw\_wb2\_v\_likp\_lips2-vbeln.  
      IF sy-subrc IS INITIAL.  
        lv\_index = sy-tabix.  
        LOOP AT gt\_wb2\_v\_likp\_lips2 INTO lw\_likp\_lips2 FROM lv\_index.  
          IF lw\_likp\_lips2-vbeln NE lw\_wb2\_v\_likp\_lips2-vbeln.  
            EXIT.  
          ENDIF.  
  
          gw\_final-taxamt    =  gw\_final-taxamt + lw\_likp\_lips2-netwr.  
          "IGST Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_likp\_lips2-knumv  
                                                   kposn = lw\_likp\_lips2-posnr\_i  
                                                   kschl = gc\_kschl\_joig  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-igst\_amt  =  gw\_final-igst\_amt + lw\_konv-kwert.  
          ENDIF.  
          "CGST Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_likp\_lips2-knumv  
                                                   kposn = lw\_likp\_lips2-posnr\_i  
                                                   kschl = gc\_kschl\_jocg  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-cgst\_amt =  gw\_final-cgst\_amt + lw\_konv-kwert.  
          ENDIF.  
          "SGST Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_likp\_lips2-knumv  
                                                   kposn = lw\_likp\_lips2-posnr\_i  
                                                   kschl = gc\_kschl\_josg  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-sgst\_amt =  gw\_final-sgst\_amt + lw\_konv-kwert.  
          ENDIF.  
          "CESS Tax amount  
          READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_likp\_lips2-knumv  
                                                   kposn = lw\_likp\_lips2-posnr\_i  
                                                   kschl = gc\_kschl\_cess  
                                                   BINARY SEARCH.  
          IF sy-subrc EQ 0.  
            gw\_final-cess\_amt =  gw\_final-cess\_amt + lw\_konv-kwert.  
          ENDIF.  
        ENDLOOP.  
      ENDIF.  
  
  
      gw\_final-invval = gw\_final-taxamt + gw\_final-cgst\_amt + gw\_final-sgst\_amt + gw\_final-igst\_amt + gw\_final-cess\_amt.  
  
      APPEND gw\_final TO gt\_final.  
      CLEAR : gw\_final.  
  
    ENDLOOP.  
  
  ENDIF.  
  
  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  FETCH\_DOMAIN\_VALUES  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM fetch\_domain\_values .  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_SUP\_TYPE'  
    TABLES  
      values\_tab      = gt\_supply\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_SUB\_TYPE'  
    TABLES  
      values\_tab      = gt\_sub\_supply\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_EDOC\_TYPE'  
    TABLES  
      values\_tab      = gt\_doctyp\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_CANCEL\_REASON'  
    TABLES  
      values\_tab      = gt\_cancel\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_T\_MODE'  
    TABLES  
      values\_tab      = gt\_t\_mode\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_EXT\_REASON'  
    TABLES  
      values\_tab      = gt\_t\_ext\_reason\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_V\_TYPE'  
    TABLES  
      values\_tab      = gt\_v\_type\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_CONSIGN\_STS'  
    TABLES  
      values\_tab      = gt\_t\_consignment\_status\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_TRANSIT\_TYPE'  
    TABLES  
      values\_tab      = gt\_t\_transit\_type\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
  
  
  CALL FUNCTION 'GET\_DOMAIN\_VALUES'  
    EXPORTING  
      domname         = 'ZDOM\_VEH\_REASON'  
    TABLES  
      values\_tab      = gt\_v\_reason\_values  
    EXCEPTIONS  
      no\_values\_found = 1  
      OTHERS          = 2.  
  IF sy-subrc = 0.  
  ENDIF.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  POPUP\_TO\_CONFIRM\_WITH\_INPUT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_VL\_PROGRAM  text  
\*      -->P\_VL\_FORMNAME  text  
\*      -->P\_VL\_POPUP\_TITLE  text  
\*      -->P\_VL\_OKPUSHBUTTON  text  
\*      -->P\_VL\_BUTTONICON  text  
\*      -->P\_LT\_SVAL  text  
\*----------------------------------------------------------------------\*  
FORM popup\_to\_confirm\_with\_input  USING    p\_vl\_program TYPE sy-repid  
                                           p\_vl\_formname TYPE char30  
                                           p\_vl\_popup\_title TYPE char30  
                                           p\_vl\_okpushbutton TYPE svalbutton-buttontext  
                                           p\_vl\_buttonicon TYPE icon-name  
                                           p\_lt\_sval TYPE ty\_sval.  
  
  
  
  CALL FUNCTION 'POPUP\_GET\_VALUES\_USER\_BUTTONS'  
    EXPORTING  
      formname          = p\_vl\_formname  
      programname       = p\_vl\_program  
      popup\_title       = p\_vl\_popup\_title  
      ok\_pushbuttontext = p\_vl\_okpushbutton  
      icon\_ok\_push      = p\_vl\_buttonicon  
    TABLES  
      fields            = p\_lt\_sval  
    EXCEPTIONS  
      error\_in\_fields   = 1  
      OTHERS            = 2.  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  CONVERT\_CURRENCY  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_GW\_FINAL\_INVVAL  text  
\*      -->P\_KURSK\_I  text  
\*      <--P\_GW\_FINAL\_INVVAL  text  
\*----------------------------------------------------------------------\*  
FORM convert\_currency  USING    iv\_value  
                                iv\_kursk  
                       CHANGING ev\_value.  
  
  DATA lv\_value TYPE wertv13.  
  
  lv\_value  = iv\_value \*   iv\_kursk.  
  ev\_value = lv\_value.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form CHANGE\_REGION\_TEXT  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM change\_region\_text .  
  FIELD-SYMBOLS <lw\_t005u> TYPE ty\_t005u.  
  LOOP AT gt\_t005u ASSIGNING <lw\_t005u>.  
    IF <lw\_t005u>-bland = '28'.  
      <lw\_t005u>-bezei = '26'.  
      MODIFY gt\_t005u FROM <lw\_t005u> TRANSPORTING bezei.  
    ENDIF.  
  
  ENDLOOP.  
  LOOP AT gt\_t005u\_s ASSIGNING <lw\_t005u>.  
    IF <lw\_t005u>-bland = '28'.  
      <lw\_t005u>-bezei = '26'.  
      MODIFY gt\_t005u\_s FROM <lw\_t005u> TRANSPORTING bezei.  
    ENDIF.  
  ENDLOOP.  
  LOOP AT gt\_t005u\_d ASSIGNING <lw\_t005u>.  
    IF <lw\_t005u>-bland = '28'.  
      <lw\_t005u>-bezei = '26'.  
      MODIFY gt\_t005u\_d FROM <lw\_t005u> TRANSPORTING bezei.  
    ENDIF.  
  ENDLOOP.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form PROCESS\_FI\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM process\_fi\_data .  
  DATA:lt\_bseg           TYPE TABLE OF ty\_bseg,  
       lv\_index          TYPE sy-tabix,  
       lw\_kna1           TYPE ty\_kna1,  
       lw\_lfa1           TYPE ty\_lfa1,  
       lw\_gstin          TYPE ty\_j\_1bbranch,  
       lw\_invrefnum      TYPE j\_1ig\_invrefnum,  
       lw\_zteinv\_details TYPE zteinv\_details,  
       lv\_internal\_date  TYPE sy-datum,  
       lw\_doctyp         TYPE ty\_doctyp,  
       lw\_einv\_hdr       TYPE zst\_einv\_api\_struct,  
       lw\_einv\_item      TYPE zst\_einv\_api\_struct\_itm.  
  
  CLEAR:gt\_bkpf,gt\_bseg,gt\_kna1,gt\_lfa1,gt\_gstin.  
  
  SELECT bukrs  
               belnr  
               gjahr  
               blart  
               bldat  
               budat  
               tcode  
               usnam  
               xblnr  
               bktxt  
               awtyp  
               awkey FROM bkpf INTO TABLE gt\_bkpf  
                          WHERE bukrs = p\_ccode  
                            AND belnr IN s\_doc  
                           AND gjahr = p\_fyear  
                           AND blart IN sl\_fkart  
                           AND budat IN s\_date.  
  IF sy-subrc = 0.  
    SELECT bukrs belnr gjahr buzei buzid bschl koart shkzg dmbtr ktosl sgtxt xauto  
              kunnr lifnr matnr werks menge meins ebeln ebelp bupla secco taxps  
              gst\_part plc\_sup hsn\_sac txgrp  
                FROM bseg INTO TABLE gt\_bseg  
                  FOR ALL ENTRIES IN gt\_bkpf  
                 WHERE bukrs = gt\_bkpf-bukrs  
                 AND belnr = gt\_bkpf-belnr  
                 AND gjahr = gt\_bkpf-gjahr  
                 AND prctr IN s\_prctr.  
    SELECT \* FROM likp INTO TABLE gt\_likp\_t  
      FOR ALL ENTRIES IN gt\_bseg  
      WHERE "bukrs = lt\_sd\_header-bukrs  
         vbeln = gt\_bseg-belnr.  
    IF gt\_likp\_t IS NOT INITIAL.  
      SELECT \* FROM dfkkbptaxnum  
      INTO TABLE gt\_dfkk  
       FOR ALL ENTRIES IN gt\_likp\_t  
        WHERE partner = gt\_likp\_t-zztrcd.  
      SELECT \* FROM zzroute  
        INTO TABLE gt\_zzroute  
        FOR ALL ENTRIES IN gt\_likp\_t  
        WHERE route = gt\_likp\_t-route.  
    ENDIF.  
    IF gt\_bseg IS NOT INITIAL.  
      lt\_bseg = gt\_bseg.  
      DELETE lt\_bseg WHERE bupla IS INITIAL.  
      SORT lt\_bseg BY bupla.  
      DELETE ADJACENT DUPLICATES FROM lt\_bseg COMPARING bupla.  
      IF lt\_bseg IS NOT INITIAL.  
  
\* GSTIN deatils  
        SELECT  bukrs branch name adrnr gstin  
           FROM j\_1bbranch  
                  INTO TABLE gt\_gstin FOR ALL ENTRIES IN lt\_bseg  
                  WHERE bukrs = lt\_bseg-bukrs  
                    AND branch = lt\_bseg-bupla.  
  
        IF sy-subrc = 0.  
          SELECT         addrnumber  
          name1  
          name2  
          city1  
          post\_code1  
          street  
          region      FROM adrc  
            INTO TABLE gt\_adrc1  
            FOR ALL ENTRIES IN gt\_gstin  
            WHERE addrnumber = gt\_gstin-adrnr.  
          IF sy-subrc IS INITIAL.  
            SELECT addrnumber smtp\_addr FROM adr6 INTO TABLE gt\_adr61  
               FOR ALL ENTRIES IN gt\_adrc1  
              WHERE addrnumber = gt\_adrc1-addrnumber.  
          ENDIF.  
        ENDIF.  
      ENDIF.  
  
\* Custoer data  
      lt\_bseg = gt\_bseg.  
      DELETE lt\_bseg WHERE kunnr IS INITIAL.  
      SORT lt\_bseg BY kunnr.  
      DELETE ADJACENT DUPLICATES FROM lt\_bseg COMPARING kunnr.  
      IF lt\_bseg IS NOT INITIAL.  
        SELECT  kunnr land1 name1 name2 ort01 pstlz  
                        regio stras telf1 adrnr stcd3  
                         FROM kna1 INTO  TABLE gt\_kna1  
                         FOR ALL ENTRIES IN lt\_bseg  
                         WHERE kunnr = lt\_bseg-kunnr.  
        IF sy-subrc EQ 0.  
          SELECT addrnumber tel\_number  FROM adrc  
            APPENDING TABLE gt\_adrc  
            FOR ALL ENTRIES IN gt\_kna1  
            WHERE addrnumber = gt\_kna1-adrnr.  
          IF sy-subrc IS INITIAL.  
            SELECT addrnumber smtp\_addr FROM adr6 APPENDING TABLE gt\_adr61  
               FOR ALL ENTRIES IN gt\_adrc1  
              WHERE addrnumber = gt\_adrc1-addrnumber.  
          ENDIF.  
        ENDIF.  
      ENDIF.  
  
\* Vendor data  
  
      lt\_bseg = gt\_bseg.  
      DELETE lt\_bseg WHERE lifnr IS INITIAL.  
      SORT lt\_bseg BY lifnr.  
      DELETE ADJACENT DUPLICATES FROM lt\_bseg COMPARING lifnr.  
      IF lt\_bseg IS NOT INITIAL.  
        SELECT \* "lifnr land1 name1 name2 name3 name4 ort01  
               "pstlz regio stras adrnr stcd3  
               FROM lfa1 INTO TABLE gt\_lfa1  
                FOR ALL ENTRIES IN lt\_bseg  
                WHERE lifnr = lt\_bseg-lifnr.  
        IF sy-subrc EQ 0.  
          SELECT addrnumber tel\_number  FROM adrc  
            APPENDING TABLE gt\_adrc  
            FOR ALL ENTRIES IN gt\_kna1  
            WHERE addrnumber = gt\_kna1-adrnr.  
          IF sy-subrc IS INITIAL.  
            SELECT addrnumber smtp\_addr FROM adr6 APPENDING TABLE gt\_adr61  
               FOR ALL ENTRIES IN gt\_adrc1  
              WHERE addrnumber = gt\_adrc1-addrnumber.  
          ENDIF.  
        ENDIF.  
      ENDIF.  
  
\* get region text  
      SELECT spras land1 bland bezei FROM t005u INTO TABLE gt\_t005u  
            WHERE spras = sy-langu  
              AND land1 = gc\_land1\_in.  
    ENDIF.  
  
  
\*\* Get e-invoice details from Z-Table  
    SELECT \* FROM zteinv\_details INTO TABLE gt\_zteinv\_details  
                                  FOR ALL ENTRIES IN gt\_bkpf  
                                  WHERE bukrs =  gt\_bkpf-bukrs  
                                  AND docno  = gt\_bkpf-belnr  
                                  AND gjahr =  gt\_bkpf-gjahr.  
    SELECT \* FROM j\_1ig\_invrefnum INTO TABLE gt\_invrefnum  
                                  FOR ALL ENTRIES IN gt\_bkpf  
                                  WHERE bukrs EQ gt\_bkpf-bukrs  
                                  AND docno EQ gt\_bkpf-belnr  
                                  AND doc\_year EQ gt\_bkpf-gjahr.  
    IF sy-subrc EQ 0.  
      SORT  gt\_invrefnum DESCENDING.  
    ENDIF.  
  
    SORT gt\_bkpf BY  bukrs belnr gjahr.  
    SORT gt\_bseg BY   bukrs belnr gjahr.  
  
  
    LOOP AT gt\_bkpf INTO gw\_bkpf.  
      gw\_final-bukrs = gw\_bkpf-bukrs.  
      gw\_final-vbeln = gw\_bkpf-belnr.  
      gw\_final-gjahr = gw\_bkpf-gjahr.  
\*      gw\_final-fkdat = gw\_bkpf-budat.  
      IF gw\_bkpf-budat IS NOT INITIAL.  
        CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
          EXPORTING  
            date\_internal            = gw\_bkpf-budat  
          IMPORTING  
            date\_external            = gw\_final-fkdat  
          EXCEPTIONS  
            date\_internal\_is\_invalid = 1  
            OTHERS                   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
        ENDIF.  
        gw\_final-fkdat\_db = gw\_bkpf-budat.  
      ENDIF.  
  
      gw\_final-fkart = gw\_bkpf-blart.  
      gw\_final-doc\_ernam = gw\_bkpf-usnam.  
      gw\_final-odnno = gw\_bkpf-xblnr.  
  
      READ TABLE gt\_bseg INTO gw\_bseg WITH KEY bukrs = gw\_bkpf-bukrs  
                                                     belnr = gw\_bkpf-belnr  
                                                     gjahr = gw\_bkpf-gjahr.  
      IF sy-subrc IS INITIAL.  
        lv\_index = sy-tabix.  
        IF gw\_bseg-kunnr IS NOT INITIAL.  
          CLEAR:lw\_kna1.  
          READ TABLE gt\_kna1 INTO lw\_kna1 WITH KEY kunnr = gw\_bseg-kunnr.  
          IF sy-subrc = 0.  
            gw\_final-kunnr = lw\_kna1-kunnr.  
            gw\_final-name1 = lw\_kna1-name1.  
            gw\_final-ship\_gstin = lw\_kna1-stcd3.  
          ENDIF.  
        ELSEIF gw\_bseg-lifnr IS NOT INITIAL.  
          CLEAR:lw\_lfa1.  
          READ TABLE gt\_lfa1 INTO lw\_lfa1 WITH KEY lifnr = gw\_bseg-lifnr.  
          IF sy-subrc = 0.  
            gw\_final-kunnr = lw\_lfa1-lifnr.  
            gw\_final-name1 = lw\_lfa1-name1.  
            gw\_final-ship\_gstin = lw\_lfa1-stcd3.  
          ENDIF.  
        ENDIF.  
        CLEAR:lw\_gstin.  
        READ TABLE gt\_gstin INTO lw\_gstin WITH KEY bukrs = gw\_bseg-bukrs  
                                                  branch = gw\_bseg-bupla.  
        IF sy-subrc = 0.  
          gw\_final-sup\_gstin = lw\_gstin-gstin.  
          gw\_final-werks = lw\_gstin-branch.  
        ENDIF.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*item Data\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
        LOOP AT gt\_bseg INTO gw\_bseg FROM lv\_index.  
          IF gw\_bseg-belnr NE gw\_bkpf-belnr .  
            EXIT.  
          ENDIF.  
  
          IF gw\_bseg-buzid  IS NOT INITIAL.  
            IF gw\_bseg-ktosl = gc\_ktosl\_jos.  
              ADD gw\_bseg-dmbtr TO gw\_final-sgst\_amt.  
            ELSEIF gw\_bseg-ktosl = gc\_ktosl\_joc.  
              ADD gw\_bseg-dmbtr TO gw\_final-cgst\_amt.  
            ELSEIF gw\_bseg-ktosl = gc\_ktosl\_joi.  
              ADD gw\_bseg-dmbtr TO gw\_final-igst\_amt.  
            ELSEIF gw\_bseg-ktosl = gc\_ktosl\_jtc.  
              ADD gw\_bseg-dmbtr TO gw\_final-tcs\_amt.  
            ENDIF.  
          ELSEIF gw\_bseg-buzid IS INITIAL.  
\*            IF gw\_bseg-koart NE 'D'.  
\*              ADD gw\_bseg-dmbtr TO gw\_final-taxamt.  
\*\*            ENDIF.  
\*            ELSEIF gw\_bseg-koart EQ 'D'OR gw\_bseg-koart EQ 'K'."gw\_bseg-ktosl = gc\_ktosl\_egk.  
\*              ADD gw\_bseg-dmbtr TO gw\_final-invval.  
\*            ENDIF.  
           IF gw\_bseg-koart EQ 'D'OR gw\_bseg-koart EQ 'K'. .  
              ADD gw\_bseg-dmbtr TO gw\_final-invval.  
\*            ENDIF.  
            ELSEIF gw\_bseg-koart NE 'D'."gw\_bseg-ktosl = gc\_ktosl\_egk.  
              ADD gw\_bseg-dmbtr TO gw\_final-taxamt.  
            ENDIF.  
          ENDIF.  
          CLEAR:gw\_bseg.  
        ENDLOOP.  
  
  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*Einvoice Data\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
        READ TABLE gt\_invrefnum INTO lw\_invrefnum WITH KEY bukrs = gw\_bkpf-bukrs  
                                                           docno = gw\_bkpf-belnr  
                                                           doc\_type = gw\_bkpf-blart  
                                                           doc\_year = gw\_bkpf-gjahr.  
        IF sy-subrc = 0.  
          gw\_final-ack\_no = lw\_invrefnum-ack\_no.  
          CLEAR:lv\_internal\_date.  
          IF lw\_invrefnum-ack\_date IS NOT INITIAL.  
            lv\_internal\_date = lw\_invrefnum-ack\_date.  
            CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
              EXPORTING  
                date\_internal            = lv\_internal\_date  
              IMPORTING  
                date\_external            = gw\_final-ack\_dt  
              EXCEPTIONS  
                date\_internal\_is\_invalid = 1  
                OTHERS                   = 2.  
            IF sy-subrc <> 0.  
              MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
            ENDIF.  
          ENDIF.  
  
          gw\_final-qr\_code = lw\_invrefnum-signed\_qrcode.  
          gw\_final-irn = lw\_invrefnum-irn.  
          gw\_final-sign\_inv = lw\_invrefnum-signed\_inv.  
          gw\_final-ernam = lw\_invrefnum-ernam.  
  
          CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
            EXPORTING  
              date\_internal            = lw\_invrefnum-erdat  
            IMPORTING  
              date\_external            = gw\_final-erdat  
            EXCEPTIONS  
              date\_internal\_is\_invalid = 1  
              OTHERS                   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
          ENDIF.  
  
          gw\_final-erzet = |{ lw\_invrefnum-erzet(2) }:{ lw\_invrefnum-erzet+2(2) }:{ lw\_invrefnum-erzet+4(2) }|."lw\_invrefnum-erzet.  
          CLEAR lv\_internal\_date.  
          IF lw\_invrefnum-cancel\_date IS NOT INITIAL.  
            lv\_internal\_date = lw\_invrefnum-cancel\_date.  
            CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
              EXPORTING  
                date\_internal            = lv\_internal\_date  
              IMPORTING  
                date\_external            = gw\_final-canc\_dt  
              EXCEPTIONS  
                date\_internal\_is\_invalid = 1  
                OTHERS                   = 2.  
            IF sy-subrc <> 0.  
              MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
            ENDIF.  
          ENDIF.  
          IF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_act.  
            gw\_final-status = 'Success'(060).  
            gw\_final-icon    = gc\_icon\_08.  
          ELSEIF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_err.  
            gw\_final-status = 'Error'(061).  
            gw\_final-icon    = gc\_icon\_0a.  
          ELSEIF lw\_invrefnum-irn\_status EQ gc\_irn\_sts\_cnl.  
            gw\_final-status = 'Cancelled'(062).  
            gw\_final-icon    = gc\_icon\_0w.  
          ENDIF.  
        ENDIF.  
        gw\_final-mod = gc\_fi.  
        CLEAR lw\_zteinv\_details.  
        READ TABLE gt\_zteinv\_details INTO lw\_zteinv\_details WITH KEY bukrs = gw\_bkpf-bukrs  
                                                                     docno = gw\_bkpf-belnr  
                                                                     doctyp = gw\_bkpf-blart  
                                                                     gjahr = gw\_bkpf-gjahr.  
  
        IF sy-subrc = 0.  
          IF gw\_final-status = 'Error'(061).  
            gw\_final-einv\_error     =  lw\_zteinv\_details-einv\_error.  
          ELSE.  
            gw\_final-eway\_error     =  ' '.  
          ENDIF.  
        ENDIF.  
  
        READ TABLE gt\_doctyp INTO lw\_doctyp WITH KEY bukrs = p\_ccode  
                                                     zmodule = gc\_fi  
                                                     fkart   = gw\_final-fkart.  
        IF sy-subrc IS INITIAL.  
          gw\_final-einv = lw\_doctyp-einv.  
          gw\_final-eway = lw\_doctyp-eway.  
        ENDIF.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*        gw\_final-invval = gw\_final-taxamt +  
\*                          gw\_final-cgst\_amt +  
\*                          gw\_final-sgst\_amt +  
\*                          gw\_final-igst\_amt +  
\*                          gw\_final-tcs\_amt.  
  
        IF gw\_final-sgst\_amt IS NOT INITIAL OR  
           gw\_final-cgst\_amt IS NOT INITIAL OR  
           gw\_final-igst\_amt IS NOT INITIAL.  
          APPEND gw\_final TO gt\_final.  
        ENDIF.  
      ENDIF.  
      CLEAR: gw\_final,gw\_bkpf.  
    ENDLOOP.  
  
  ENDIF.  
ENDFORM.

INCLUDE zrep\_einv\_eway\_report\_statuo01.

\*----------------------------------------------------------------------\*  
\*\*\*INCLUDE ZREP\_EINV\_EWAY\_REPORT\_STATUO01.  
\*----------------------------------------------------------------------\*  
\*&---------------------------------------------------------------------\*  
\*&      Module  STATUS\_9000  OUTPUT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
MODULE status\_9000 OUTPUT.  
  SET PF-STATUS '9000'.  
  SET TITLEBAR  '9000'.  
  
  DATA: lt\_variant TYPE  disvariant,  
        lw\_layout  TYPE lvc\_s\_layo.  
  DATA:o\_eventreceiver  TYPE REF TO lcl\_event.  
  CLEAR lw\_layout.  
  
  lw\_layout-zebra = space.  
  lw\_layout-cwidth\_opt = abap\_true.  
  lw\_layout-sel\_mode = gc\_lay\_mode\_a."'A'.  
  
\* Create Controls  
  IF gref\_alv\_container IS NOT BOUND.  
    CREATE OBJECT gref\_alv\_container  
      EXPORTING  
        container\_name = 'CUST\_ALV'.  
  
    CREATE OBJECT dg\_dyndoc\_id  
      EXPORTING  
        style = 'ALV\_GRID'.  
\* Create Splitter for custom\_container  
    CREATE OBJECT dg\_splitter  
      EXPORTING  
        parent  = gref\_alv\_container  
        rows    = 2  
        columns = 1.  
\* Split the custom\_container to two containers and move the reference  
\* to receiving containers g\_parent\_html and g\_parent\_grid  
    "i am allocating the space for grid and top of page  
    CALL METHOD dg\_splitter->get\_container  
      EXPORTING  
        row       = 1  
        column    = 1  
      RECEIVING  
        container = dg\_parent\_html.  
  
    CALL METHOD dg\_splitter->get\_container  
      EXPORTING  
        row       = 2  
        column    = 1  
      RECEIVING  
        container = dg\_parent\_grid.  
  
    "you can set the height of it  
\* Set height for g\_parent\_html  
    CALL METHOD dg\_splitter->set\_row\_height  
      EXPORTING  
        id     = 1  
        height = 20.  
    "from here as usual..you need to specify parent as splitter part  
    "which we alloted for grid  
    CREATE OBJECT gref\_alv\_grid  
      EXPORTING  
        i\_parent = dg\_parent\_grid.  
  
    IF gref\_alv\_container IS BOUND.  
  
\*  Populate Field Catalog  
      PERFORM get\_fieldcatalog.  
  
      CREATE OBJECT go\_obj.  
      IF go\_obj IS BOUND.  
        SET HANDLER : go\_obj->toolbar FOR gref\_alv\_grid,  
                      go\_obj->user\_command FOR gref\_alv\_grid,  
                      go\_obj->menu FOR gref\_alv\_grid,  
                      go\_obj->handle\_top\_of\_page FOR gref\_alv\_grid,  
                      go\_obj->handle\_hotspot\_click FOR   gref\_alv\_grid.  
  
  
        lt\_variant-report = sy-repid.  
        lt\_variant-username = sy-uname.  
        lt\_variant-handle = 'GRID'.  
  
  
        CALL METHOD gref\_alv\_grid->set\_table\_for\_first\_display  
          EXPORTING  
            is\_layout       = lw\_layout  
            is\_variant      = lt\_variant  
            i\_save          = gc\_alv\_save\_a  
          CHANGING  
            it\_outtab       = gt\_final[]  
            it\_fieldcatalog = gt\_fieldcat[].  
      ENDIF.  
    ENDIF.  
  
  ELSE.  
    CALL METHOD gref\_alv\_grid->refresh\_table\_display  
      EXCEPTIONS  
        finished = 1  
        OTHERS   = 2.  
    IF sy-subrc <> 0.  
      MESSAGE 'Table refresh error'(012) TYPE 'I'.  
    ENDIF.  
  
  ENDIF.  
  CALL METHOD gref\_alv\_grid->list\_processing\_events  
    EXPORTING  
      i\_event\_name = 'TOP\_OF\_PAGE'  
      i\_dyndoc\_id  = dg\_dyndoc\_id.  
  
ENDMODULE.  
  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  GET\_FIELDCATALOG  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM get\_fieldcatalog .  
  
  REFRESH: gt\_fieldcat.  
\* Document  
  PERFORM fill\_fieldcatalog USING 'E-Invoice Status'(086) 'ICON' 'GT\_FINAL' '7' 'X'.  
  PERFORM fill\_fieldcatalog USING 'E-Way Status'(102)  'EWAY\_ICON' 'GT\_FINAL' '7' 'X'.  
  PERFORM fill\_fieldcatalog USING 'Supplier GSTIN'(063) 'SUP\_GSTIN' 'GT\_FINAL' '20' 'X'.  
  PERFORM fill\_fieldcatalog USING 'Plant'(064) 'WERKS' 'GT\_FINAL' '6' 'X'.  
  PERFORM fill\_fieldcatalog USING 'Document Number'(065) 'VBELN' 'GT\_FINAL' '15' 'X'.  
  PERFORM fill\_fieldcatalog USING 'Document Date'(066) 'FKDAT' 'GT\_FINAL' '15' 'X'.  
  PERFORM fill\_fieldcatalog USING 'Doc Type'(067) 'FKART' 'GT\_FINAL' '10' ''.  
  PERFORM fill\_fieldcatalog USING 'Created By'(068) 'DOC\_ERNAM' 'GT\_FINAL' '12' ''.  
  PERFORM fill\_fieldcatalog USING 'GST Invoice No'(069) 'ODNNO' 'GT\_FINAL' '16' ''.  
  IF p\_mod = gc\_sd.  
    PERFORM fill\_fieldcatalog USING 'Customer'(070) 'KUNNR' 'GT\_FINAL' '12' ''.  
    PERFORM fill\_fieldcatalog USING 'Customer Name'(071) 'NAME1' 'GT\_FINAL' '20' ''.  
    PERFORM fill\_fieldcatalog USING 'Customer GSTIN'(117) 'SHIP\_GSTIN' 'GT\_FINAL' '20' ''.  
  ELSEIF p\_mod = gc\_mm.  
    PERFORM fill\_fieldcatalog USING  'Vendor'(072) 'KUNNR' 'GT\_FINAL' '12' ''.  
    PERFORM fill\_fieldcatalog USING  'Vendor Name'(073) 'NAME1' 'GT\_FINAL' '20' ''.  
  ENDIF.  
  PERFORM fill\_fieldcatalog USING 'Ship To'(074) 'SHIP\_TO' 'GT\_FINAL' '12' ''.  
  PERFORM fill\_fieldcatalog USING 'Ship To Name'(075) 'SHIP\_TO\_NAME' 'GT\_FINAL' '20' ''.  
\* Values  
  PERFORM fill\_fieldcatalog USING 'Invoice Value'(076) 'INVVAL' 'GT\_FINAL' '15' ''.  
  PERFORM fill\_fieldcatalog USING 'Taxable Value'(077) 'TAXAMT' 'GT\_FINAL' '15' ''.  
  PERFORM fill\_fieldcatalog USING 'CGST'(078) 'CGST\_AMT' 'GT\_FINAL' '15' ''.  
  PERFORM fill\_fieldcatalog USING 'SGST'(079) 'SGST\_AMT' 'GT\_FINAL' '15' ''.  
  PERFORM fill\_fieldcatalog USING 'IGST'(080) 'IGST\_AMT' 'GT\_FINAL' '15' ''.  
  PERFORM fill\_fieldcatalog USING 'CESS'(081) 'CESS\_AMT' 'GT\_FINAL' '15' ''.  
\* E-Invoice  
  READ TABLE gt\_doctyp TRANSPORTING NO FIELDS WITH KEY einv = abap\_true.  
  IF sy-subrc IS INITIAL.  
    PERFORM fill\_fieldcatalog USING 'Ack No'(082) 'ACK\_NO' 'GT\_FINAL' '15' ''.  
    PERFORM fill\_fieldcatalog USING 'Ack Date'(083) 'ACK\_DT' 'GT\_FINAL' '15' ''.  
    PERFORM fill\_fieldcatalog USING 'IRN'(084) 'IRN' 'GT\_FINAL' '25' ''.  
    PERFORM fill\_fieldcatalog USING 'Cancel Date'(085) 'CANC\_DT' 'GT\_FINAL' '12' ''.  
    PERFORM fill\_fieldcatalog USING 'Error description'(087) 'EINV\_ERROR' 'GT\_FINAL' '50' ''.  
    PERFORM fill\_fieldcatalog USING 'Created By'(088) 'ERNAM' 'GT\_FINAL' '12' ''.  
    PERFORM fill\_fieldcatalog USING 'Created Date'(089) 'ERDAT' 'GT\_FINAL' '10' ''.  
    PERFORM fill\_fieldcatalog USING 'Created Time'(090) 'ERZET' 'GT\_FINAL' '8' ''.  
  ENDIF.  
  
\* E-Way Bill  
  READ TABLE gt\_doctyp TRANSPORTING NO FIELDS WITH KEY eway = abap\_true.  
  IF sy-subrc IS INITIAL.  
  
    PERFORM fill\_fieldcatalog USING 'Transporter ID'(091) 'T\_ID' 'GT\_FINAL' '15' ''.  
    PERFORM fill\_fieldcatalog USING 'Transporter Name'(092) 'T\_NAME' 'GT\_FINAL' '20' ''.  
    PERFORM fill\_fieldcatalog USING 'Mode'(093)          'T\_MODE' 'GT\_FINAL' '5' ''.  
    PERFORM fill\_fieldcatalog USING 'Document No'(094)   'T\_DOC\_NO' 'GT\_FINAL' '15' ''.  
    PERFORM fill\_fieldcatalog USING 'Document Date'(095) 'T\_DATE' 'GT\_FINAL' '12' ''.  
\*    PERFORM fill\_fieldcatalog USING 'Distance in KM' 'T\_DISTANCE' 'GT\_FINAL' '5' ''.  
    PERFORM fill\_fieldcatalog USING 'Vehicle No'(096)   'V\_NUMBER' 'GT\_FINAL' '20' ''.  
    PERFORM fill\_fieldcatalog USING 'Vehicle Type'(097) 'V\_TYPE' 'GT\_FINAL' '20' ''.  
    PERFORM fill\_fieldcatalog USING 'E-Waybill No'(098) 'EWAY\_NUM' 'GT\_FINAL' '12' ''.  
    PERFORM fill\_fieldcatalog USING 'E-Way date'(099)   'EWAY\_DATE' 'GT\_FINAL' '15' ''.  
    PERFORM fill\_fieldcatalog USING 'Valid Upto'(100)      'EWAY\_V\_TO' 'GT\_FINAL' '10' ''.  
    PERFORM fill\_fieldcatalog USING 'Cancel Date'(101)  'EWAY\_CANC\_DT' 'GT\_FINAL' '10' ''.  
    PERFORM fill\_fieldcatalog USING 'Error description'(103) 'EWAY\_ERROR' 'GT\_FINAL' '50' ''.  
    PERFORM fill\_fieldcatalog USING 'Created By'(104)   'EWAY\_ERNAM' 'GT\_FINAL' '12' ''.  
    PERFORM fill\_fieldcatalog USING 'Created Date'(105) 'EWAY\_ERDAT' 'GT\_FINAL' '10' ''.  
    PERFORM fill\_fieldcatalog USING 'Created Time'(106) 'EWAY\_ERZET' 'GT\_FINAL' '8' ''.  
  
  ENDIF.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  FILL\_FIELDCATALOG  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_0707   text  
\*      -->P\_0708   text  
\*      -->P\_0709   text  
\*      -->P\_0710   text  
\*----------------------------------------------------------------------\*  
FORM fill\_fieldcatalog  USING    VALUE(p\_0707)  
                                 VALUE(p\_0708)  
                                 VALUE(p\_0709)  
                                 VALUE(p\_0710)  
                                 VALUE(p\_0711).  
  
  
  DATA: lw\_fcat TYPE lvc\_s\_fcat.  
  
  CLEAR: lw\_fcat.  
  lw\_fcat-reptext    = p\_0707.  
  lw\_fcat-fieldname  = p\_0708.  
  lw\_fcat-ref\_table  = p\_0709.  
  lw\_fcat-outputlen  = p\_0710.  
  lw\_fcat-key        = p\_0711.  
  IF  lw\_fcat-fieldname  = 'ICON'.  
    lw\_fcat-icon = 'X'.  
  ENDIF.  
  IF lw\_fcat-fieldname EQ 'INVVAL' OR  
     lw\_fcat-fieldname EQ 'TAXAMT' OR  
     lw\_fcat-fieldname EQ 'CGST\_AMT' OR  
     lw\_fcat-fieldname EQ 'SGST\_AMT' OR  
     lw\_fcat-fieldname EQ 'IGST\_AMT' OR  
     lw\_fcat-fieldname EQ 'CESS\_AMT'.  
  
    lw\_fcat-do\_sum = 'X'.  
  ENDIF.  
  
  IF  lw\_fcat-fieldname  = 'VBELN'.  
    lw\_fcat-hotspot = 'X'.  
  ENDIF.  
  
  APPEND lw\_fcat TO gt\_fieldcat.  
  
ENDFORM.  
  
\*&---------------------------------------------------------------------\*  
\*&      Module  STATUS\_9001  OUTPUT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
MODULE status\_9001 OUTPUT.  
  SET PF-STATUS '9001\_STATUS'.  
  SET TITLEBAR '9001\_TITLE'.  
  
  PERFORM set\_distance.  
  PERFORM set\_port\_det.  
  
  
ENDMODULE.  
  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  SET\_DISTANCE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM set\_distance .  
  DATA:lw\_index LIKE LINE OF gt\_index,  
       lv\_val   TYPE xfeld.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  IF lv\_val IS INITIAL.  
    LOOP AT  gt\_index INTO lw\_index.  
      READ TABLE gt\_final ASSIGNING <gw\_final> INDEX lw\_index-index.  
      IF sy-subrc IS INITIAL AND <gw\_final>-t\_distance IS INITIAL.  
        IF <gw\_final>-t\_distance IS NOT INITIAL.  
          zst\_einv\_api\_struct-transportation\_distance = <gw\_final>-t\_distance.  
        ENDIF.  
      ENDIF.  
      zst\_einv\_api\_struct-transporter\_id              = <gw\_final>-t\_id.  
      zst\_einv\_api\_struct-transporter\_name            = <gw\_final>-t\_name.  
  
      zst\_einv\_api\_struct-transportation\_mode         = <gw\_final>-t\_mode.  
      zst\_einv\_api\_struct-transportation\_distance     = <gw\_final>-t\_distance.  
      zst\_einv\_api\_struct-transporter\_document\_number = <gw\_final>-t\_doc\_no.  
  
      CONCATENATE <gw\_final>-t\_date+6(4)  
                  <gw\_final>-t\_date+3(2)  
                  <gw\_final>-t\_date+0(2)  
                  INTO zst\_einv\_api\_struct-transporter\_document\_date.  
  
      zst\_einv\_api\_struct-vehicle\_number              = <gw\_final>-v\_number.  
  
  
      zst\_einv\_api\_struct-vehicle\_type              = <gw\_final>-v\_type.  
      IF zst\_einv\_api\_struct-vehicle\_type IS INITIAL.  
        zst\_einv\_api\_struct-vehicle\_type = gc\_v\_type\_1.  
      ENDIF.  
      zst\_einv\_api\_struct-reason\_code\_for\_vehicle\_updati             = <gw\_final>-v\_reason\_code.  
      zst\_einv\_api\_struct-reason\_for\_vehicle\_updation            = <gw\_final>-v\_reason.  
      zst\_einv\_api\_struct-remaining\_distance        = <gw\_final>-t\_r\_distance.  
      zst\_einv\_api\_struct-extend\_validity\_reason    = <gw\_final>-t\_ext\_valid\_reason.  
      zst\_einv\_api\_struct-extend\_remarks            = <gw\_final>-t\_ext\_valid\_remarks.  
      zst\_einv\_api\_struct-from\_pincode              = <gw\_final>-t\_from\_pin.  
      zst\_einv\_api\_struct-consignment\_status        = <gw\_final>-t\_consignment\_status.  
      zst\_einv\_api\_struct-transit\_type              = <gw\_final>-t\_transit\_type.  
      zst\_einv\_api\_struct-address\_line1             = <gw\_final>-t\_address1.  
      zst\_einv\_api\_struct-address\_line2             = <gw\_final>-t\_address2.  
      zst\_einv\_api\_struct-address\_line3             = <gw\_final>-t\_address3.  
      zst\_einv\_api\_struct-zport\_code                     = <gw\_final>-zport\_code.  
  
    ENDLOOP.  
  
  ENDIF.  
ENDFORM.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  SET\_PORT\_DET  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM set\_port\_det .  
  
  DATA:lt\_list TYPE vrm\_values,  
       lw\_list TYPE vrm\_value,  
       lt\_port TYPE TABLE OF ty\_export,  
       ls\_port TYPE ty\_export.  
  
  CLEAR:lt\_list,lt\_port.  
  
  SELECT zport\_code  zport land1 zport\_state  
         zport\_address1 zport\_address2  
         zport\_place zport\_pincode  
        FROM ztedoc\_export INTO TABLE lt\_port WHERE land1 = gc\_land1\_in..  
  
  IF sy-subrc IS INITIAL.  
    LOOP AT lt\_port INTO ls\_port.  
      lw\_list-key = ls\_port-zport\_code.  
      lw\_list-text = ls\_port-zport.  
      APPEND lw\_list TO lt\_list.  
    ENDLOOP.  
  
    CALL FUNCTION 'VRM\_SET\_VALUES'  
      EXPORTING  
        id              = 'ZST\_EINV\_API\_STRUCT-ZPORT\_CODE'  
        values          = lt\_list  
      EXCEPTIONS  
        id\_illegal\_name = 1  
        OTHERS          = 2.  
  
  ENDIF.  
  
ENDFORM.

  INCLUDE zrep\_einv\_eway\_report\_user\_i01.

\*----------------------------------------------------------------------\*  
\*\*\*INCLUDE ZREP\_EINV\_EWAY\_REPORT\_USER\_I01.  
\*----------------------------------------------------------------------\*  
\*&---------------------------------------------------------------------\*  
\*&      Module  USER\_COMMAND\_9000  INPUT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
MODULE user\_command\_9000 INPUT.  
  
  CASE sy-ucomm.  
  
    WHEN 'BACK' OR 'CANCEL' OR 'EXIT'.  
      PERFORM exit\_program.  
  ENDCASE.  
  
ENDMODULE.  
\*&---------------------------------------------------------------------\*  
\*&      Form  CHECK\_SELECTED\_DATA  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM check\_selected\_data CHANGING  p\_val.  
  
  DATA:  
       lv\_line  TYPE i.  
  
  CLEAR:gt\_index,gt\_row.  
  
  
  CALL METHOD gref\_alv\_grid->get\_selected\_rows  
    IMPORTING  
      et\_index\_rows = gt\_index  
      et\_row\_no     = gt\_row.  
  
  
  
  DESCRIBE TABLE gt\_row LINES lv\_line.  
  IF lv\_line IS INITIAL.  
  
    p\_val = abap\_true.  
    MESSAGE 'Please select atleast one document'(013) TYPE 'I'.  
  
    EXIT.  
  
  ENDIF.  
  
  
  
  
ENDFORM.  
FORM set\_selected\_data.  
  IF gt\_index IS NOT INITIAL.  
    CALL METHOD gref\_alv\_grid->set\_selected\_rows  
      EXPORTING  
        it\_index\_rows = gt\_index  
        it\_row\_no     = gt\_row.  
  ENDIF.  
  
ENDFORM.  
  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  EINVOICE\_GENERATE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM einvoice\_generate .  
  
  DATA: lw\_index            TYPE lvc\_s\_row,  
        lv\_val              TYPE xfeld,  
        lw\_token            TYPE string,  
        lw\_return           TYPE string,  
        wa\_return           TYPE string,  
        lt\_messages         TYPE bapiret2\_t,  
        lw\_message          TYPE bapiret2,  
        lt\_show\_message     TYPE esp1\_message\_tab\_type,  
        lw\_show\_message     LIKE LINE OF lt\_show\_message,  
        lt\_invref           TYPE TABLE OF j\_1ig\_invrefnum,  
        lw\_invref           TYPE  j\_1ig\_invrefnum,  
        lt\_einv\_details     TYPE TABLE OF zteinv\_details,  
        lw\_einv\_details     TYPE zteinv\_details,  
        lw\_kna1             TYPE ty\_kna1,  
        lw\_t005u            TYPE ty\_t005u,  
        lw\_adrc             TYPE ty\_adrc,  
        lw\_adr6             TYPE ty\_adr6,  
        lw\_export           TYPE ty\_export,  
        lw\_vbfa             TYPE ty\_vbfa,  
        lw\_wb2\_v\_vbak\_vbap2 TYPE ty\_wb2\_v\_vbak\_vbap2,  
        lv\_reverse\_charge   TYPE i VALUE 1,  
\*Seller Details  
        lw\_t001w            TYPE ty\_t001w1,  
        lw\_adrc\_s           TYPE ty\_adrc,  
        lw\_t005u\_s          TYPE ty\_t005u,  
        lw\_adr6\_s           TYPE ty\_adr6,  
        lw\_wb2\_v\_vbrk\_vbrp2 TYPE ty\_wb2\_v\_vbrk\_vbrp2,  
        lw\_bkpfm            TYPE ty\_bkpf1,  
        lw\_bseg             TYPE ty\_bseg1,  
        lw\_marc             TYPE ty\_marc,  
        lv\_export           TYPE xfeld,  
\*Dispatch and Ship Details  
        lw\_likp             TYPE ty\_likp,  
        lw\_kna1\_d           TYPE ty\_kna1,  
        lw\_t005u\_d          TYPE ty\_t005u,  
        lw\_adrc\_d           TYPE ty\_adrc,  
        lt\_lines            TYPE STANDARD TABLE OF tline,  
        lw\_lines            TYPE tline,  
        lv\_index            TYPE sy-tabix,  
        lw\_einv\_api\_hdr     TYPE zst\_einv\_api\_struct,  
        lw\_einv\_api\_itm     TYPE zst\_einv\_api\_struct\_itm,  
        ls\_doctyp           TYPE zveinv\_doctyp,  
        ls\_values           TYPE dd07v,  
        lt\_api\_hdr          TYPE ztt\_einv\_api\_struct,  
        lt\_api\_itm          TYPE ztt\_einv\_api\_struct\_itm,  
        lv\_answer           TYPE c,  
        lw\_usrgstin         TYPE ty\_api,  
        lw\_vbpa             TYPE ty\_vbpa,  
        lw\_konv             TYPE ty\_konv,  
        lw\_gstrate          TYPE kbetr,  
        ls\_doc\_uom          TYPE ty\_doc\_uom,  
        lv\_output\_uom       TYPE meins,  
        lw\_doctyp           TYPE ty\_doctyp,  
        lw\_tvarvc           TYPE ty\_tvarvc,  
        lv\_tdname           TYPE thead-tdname,  
        lv\_unit\_price       TYPE dmbtr,  
        lv\_exp\_curr         TYPE xfeld,  
        lv\_dmbtr            TYPE dmbtr,  
        lv\_rate             TYPE dmbtr,  
        lv\_item\_amt         TYPE wertv13,  
        lv\_roundoff         TYPE dmbtr.  
  
  
  
  DATA: lv\_total\_assessable\_value    TYPE wertv13,  
        lv\_total\_cgst\_value          TYPE wertv13,  
        lv\_total\_sgst\_value          TYPE wertv13,  
        lv\_total\_igst\_value          TYPE wertv13,  
        lv\_total\_invoice\_value       TYPE wertv13,  
        lv\_total\_cess\_value          TYPE wertv13,  
        lv\_total\_cess\_nonadvol\_value TYPE wertv13.  
  
  DATA: lv\_supply\_type     TYPE char10 VALUE 'O',  
        lv\_document\_type   TYPE char10 VALUE 1,  
        lv\_sub\_supply\_type TYPE char10 VALUE 1,  
        lv\_document\_number TYPE i VALUE 2,  
        lv\_date            TYPE sy-datum.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  
  CLEAR: lt\_messages,lt\_invref,lt\_api\_hdr,lt\_api\_itm,gt\_einv\_api\_hdr,gt\_einv\_api\_itm.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
    PERFORM call\_popup USING   TEXT-t20 TEXT-t21  
                       CHANGING lv\_answer.  
  
    IF lv\_answer = '1'.  
      LOOP AT  gt\_index INTO lw\_index.  
        CLEAR:lw\_wb2\_v\_vbrk\_vbrp2.  
  
        READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND gw\_final-irn IS INITIAL AND gw\_final-einv = abap\_true.  
          IF p\_mod = gc\_sd.  
            READ TABLE gt\_wb2\_v\_vbrk\_vbrp2 INTO lw\_wb2\_v\_vbrk\_vbrp2 WITH KEY vbeln = gw\_final-vbeln.  
            IF sy-subrc = 0.  
              IF lw\_wb2\_v\_vbrk\_vbrp2-fksto NE abap\_true.  
                lv\_index = sy-tabix.  
  
  
                lv\_total\_assessable\_value    = 0.  
                lv\_total\_cgst\_value          = 0.  
                lv\_total\_sgst\_value          = 0.  
                lv\_total\_igst\_value          = 0.  
                lv\_total\_invoice\_value       = 0.  
                lv\_total\_cess\_value          = 0.  
                lv\_total\_cess\_nonadvol\_value = 0.  
  
                CLEAR:lw\_einv\_api\_hdr-total\_assessable\_value,lw\_einv\_api\_hdr-total\_cgst\_value,lw\_einv\_api\_hdr-total\_sgst\_value,lw\_einv\_api\_hdr-total\_igst\_value,  
                                     lw\_einv\_api\_hdr-total\_invoice\_value,lw\_einv\_api\_hdr-total\_cess\_value,lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value.  
  
                CLEAR: lv\_total\_assessable\_value,lv\_total\_igst\_value,  lv\_total\_cgst\_value,lv\_total\_sgst\_value, lv\_total\_cess\_value,lv\_total\_cess\_nonadvol\_value, lv\_total\_invoice\_value.  
                LOOP AT gt\_wb2\_v\_vbrk\_vbrp2 INTO lw\_wb2\_v\_vbrk\_vbrp2 FROM lv\_index.  
                  IF lw\_wb2\_v\_vbrk\_vbrp2-vbeln NE gw\_final-vbeln.  
                    EXIT.  
                  ENDIF.  
  
                  lw\_einv\_api\_hdr-vbeln  = lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
                  lw\_einv\_api\_hdr-bukrs  = lw\_wb2\_v\_vbrk\_vbrp2-bukrs.  
                  lw\_einv\_api\_hdr-doc\_year = gw\_final-gjahr.  
                  lw\_einv\_api\_hdr-doc\_type = lw\_wb2\_v\_vbrk\_vbrp2-fkart.  
  
\*Data to send to the API IN JSON Scheme  
                  READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_usrgstin.  
                  IF sy-subrc IS INITIAL.  
                    lw\_einv\_api\_hdr-user\_gstin   = lw\_usrgstin-apiprov.  
                  ELSE.  
                    lw\_einv\_api\_hdr-user\_gstin   =  gw\_final-sup\_gstin.  
                  ENDIF.  
                  CONCATENATE  '"' lw\_einv\_api\_hdr-user\_gstin '"' INTO lw\_einv\_api\_hdr-user\_gstin.  
  
                  lw\_einv\_api\_hdr-supply\_type   = gc\_sup\_type\_b2b.  
                  CASE lv\_reverse\_charge.  
                    WHEN 1.  
                      lw\_einv\_api\_hdr-charge\_type = gc\_charge\_n.  
                    WHEN 2.  
                      lw\_einv\_api\_hdr-charge\_type = gc\_charge\_y.  
                  ENDCASE.  
                  lw\_einv\_api\_hdr-ecommerce\_gstin = '""'.  
  
  
                  CASE lv\_document\_number.  
                    WHEN 1.   "Billing Number as same as ODN  
                      lw\_einv\_api\_hdr-document\_number = lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
                    WHEN 2.   "ODN  
                      lw\_einv\_api\_hdr-document\_number = lw\_wb2\_v\_vbrk\_vbrp2-xblnr.  
                  ENDCASE.  
                  SHIFT lw\_einv\_api\_hdr-document\_number LEFT DELETING LEADING '0'.  
  
                  IF lw\_wb2\_v\_vbrk\_vbrp2-fkdat IS NOT INITIAL.  
                    CONCATENATE  lw\_wb2\_v\_vbrk\_vbrp2-fkdat+6(2) '/'  
                                 lw\_wb2\_v\_vbrk\_vbrp2-fkdat+4(2) '/'  
                                 lw\_wb2\_v\_vbrk\_vbrp2-fkdat+0(4)  
                            INTO lw\_einv\_api\_hdr-document\_date.                                "Required         "Document Date "[2][0][1-2][0-9]-[0-1][0-9]-[0-3][0-9]"  Y  
                  ENDIF.  
  
                  READ TABLE gt\_doctyp INTO lw\_doctyp WITH KEY fkart = lw\_wb2\_v\_vbrk\_vbrp2-fkart  
                                                               bukrs = lw\_wb2\_v\_vbrk\_vbrp2-bukrs  
                                                               zmodule = gc\_sd.  
                  IF sy-subrc = 0.  
                    lw\_einv\_api\_hdr-document\_type = lw\_doctyp-edoc\_type.  
                  ELSE.  
                    lw\_einv\_api\_hdr-document\_type = gc\_docty\_inv.  
                  ENDIF.  
  
  
                  CONCATENATE '"' lw\_einv\_api\_hdr-document\_type '"' INTO  lw\_einv\_api\_hdr-document\_type.  
                  CONCATENATE '"' lw\_einv\_api\_hdr-document\_number '"' INTO  lw\_einv\_api\_hdr-document\_number.  
                  CONCATENATE '"' lw\_einv\_api\_hdr-document\_date '"' INTO  lw\_einv\_api\_hdr-document\_date.  
  
\*\*Begin of seller\_details  
                  CLEAR: lw\_t001w, lw\_adrc\_s, lw\_adr6\_s.  
                  READ TABLE gt\_t001w INTO lw\_t001w WITH  KEY werks = lw\_wb2\_v\_vbrk\_vbrp2-werks\_i BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    READ TABLE gt\_adrc\_s INTO lw\_adrc\_s WITH  KEY addrnumber = lw\_t001w-adrnr.  
                    READ TABLE gt\_adr6\_s INTO lw\_adr6\_s WITH  KEY addrnumber = lw\_adrc\_s-addrnumber.  
                    READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_usrgstin.  
                    IF sy-subrc IS INITIAL.  
                      lw\_einv\_api\_hdr-gstin\_s   = lw\_usrgstin-apiprov.  
                    ELSE.  
                      lw\_einv\_api\_hdr-gstin\_s   =  gw\_final-sup\_gstin.  
                    ENDIF.  
                    CONCATENATE '"' lw\_einv\_api\_hdr-gstin\_s '"' INTO lw\_einv\_api\_hdr-gstin\_s.  
  
                    CONCATENATE '"'   lw\_t001w-name1  '"' INTO   lw\_einv\_api\_hdr-legal\_name\_s   .     "Legalname   Y       "Required  
                    CONCATENATE '"'   lw\_t001w-name1  '"' INTO   lw\_einv\_api\_hdr-trade\_name\_s    .    "Tradename   N  
                    CONCATENATE '"'   lw\_t001w-name2  '"' INTO   lw\_einv\_api\_hdr-address1\_s      .   "Building no.    y     "Required  
                    CONCATENATE '"'   lw\_t001w-stras  '"' INTO   lw\_einv\_api\_hdr-address2\_s               .  "Building name   N  
                    CONCATENATE '"'   lw\_t001w-ort01  '"' INTO   lw\_einv\_api\_hdr-location\_s       .  "Location    Y         "Required  
  
                    lw\_einv\_api\_hdr-pincode\_s =  lw\_t001w-pstlz    .  "Pincode   Y           "Required  
  
                    CLEAR:lw\_t005u\_s.  
                    READ TABLE gt\_t005u\_s INTO lw\_t005u\_s WITH  KEY bland = lw\_t001w-regio.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE lw\_t005u\_s-bezei TO UPPER CASE.  
                      lw\_einv\_api\_hdr-state\_code\_s = lw\_t005u\_s-bezei . "State code    Y        "Required  
                    ENDIF.  
                    CONCATENATE '"'  lw\_einv\_api\_hdr-state\_code\_s  '"' INTO lw\_einv\_api\_hdr-state\_code\_s.  
                    CONCATENATE '"'  lw\_adrc\_s-tel\_number+0(10)  '"' INTO  lw\_einv\_api\_hdr-phone\_number\_s.        "Phone or Mobile No.   N  
                    CONCATENATE '"'    lw\_adr6\_s-smtp\_addr '"' INTO    lw\_einv\_api\_hdr-email\_s.             "Email ID                                                  "Email-Id    N  
                  ENDIF.  
\*\*End of seller\_details  
  
\*\*Begin of buyer\_details  
                  CLEAR:lw\_kna1, lw\_adrc, lw\_adr6.  
                  READ TABLE gt\_vbpa INTO lw\_vbpa WITH KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                           parvw = gc\_parvw\_re  
                                                           BINARY SEARCH.  
                  IF sy-subrc = 0.  
                    READ TABLE gt\_kna1 INTO lw\_kna1 WITH  KEY kunnr = lw\_vbpa-kunnr BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      READ TABLE gt\_adrc INTO lw\_adrc WITH  KEY addrnumber = lw\_kna1-adrnr.  
                      READ TABLE gt\_adr6 INTO lw\_adr6 WITH  KEY addrnumber = lw\_adrc-addrnumber.  
                      CONCATENATE '"'  lw\_kna1-stcd3 '"' INTO          lw\_einv\_api\_hdr-gstin\_b .  
                      CONCATENATE '"'  lw\_kna1-name1 '"' INTO          lw\_einv\_api\_hdr-legal\_name\_b.  
                      CONCATENATE '"' lw\_kna1-name1 '"' INTO lw\_einv\_api\_hdr-trade\_name\_b  .  
                      IF lw\_kna1-name1 IS NOT INITIAL.  
                        lw\_einv\_api\_hdr-address1\_b = lw\_adrc-str\_suppl1.  
                      ENDIF.  
                      IF lw\_einv\_api\_hdr-address1\_b IS INITIAL.  
                        lw\_einv\_api\_hdr-address1\_b = lw\_adrc-str\_suppl2.  
                      ENDIF.  
                      CONCATENATE '"' lw\_einv\_api\_hdr-address1\_b '"' INTO lw\_einv\_api\_hdr-address1\_b .  
                      CONCATENATE '"' lw\_kna1-name2 '"' INTO  lw\_einv\_api\_hdr-address2\_b.  
                      CONCATENATE '"'  lw\_kna1-ort01 '"'  INTO           lw\_einv\_api\_hdr-location\_b.  
                      lw\_einv\_api\_hdr-pincode\_b  = lw\_kna1-pstlz.  
  
                      lw\_einv\_api\_hdr-place\_of\_supply\_b = lw\_kna1-regio.  
                      CONCATENATE '"' lw\_einv\_api\_hdr-place\_of\_supply\_b '"' INTO lw\_einv\_api\_hdr-place\_of\_supply\_b.  
                      CLEAR:lw\_t005u.  
                      READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_kna1-regio.  
                      IF sy-subrc EQ 0.  
                        TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                        lw\_einv\_api\_hdr-state\_code\_b = lw\_t005u-bezei.  
                        "State code    Y  
                      ENDIF.  
                      CONCATENATE '"' lw\_einv\_api\_hdr-state\_code\_b  '"'  INTO lw\_einv\_api\_hdr-state\_code\_b .  
                      lw\_einv\_api\_hdr-place\_of\_supply\_b   =   lw\_einv\_api\_hdr-state\_code\_b.  
                      CONCATENATE '"' lw\_adrc-tel\_number+0(10) '"' INTO   lw\_einv\_api\_hdr-phone\_number\_b.                                         "Phone or Mobile No.   N  
                      CONCATENATE '"'   lw\_adr6-smtp\_addr '"' INTO  lw\_einv\_api\_hdr-email\_b.  
  
                      IF lw\_t001w-land1 NE lw\_kna1-land1.  
                        lw\_einv\_api\_hdr-gstin\_b = gc\_gstin\_urp.  
                        READ TABLE gt\_export INTO lw\_export WITH KEY zport\_code = gw\_final-zport\_code.  
                        IF sy-subrc IS INITIAL.  
                          CONCATENATE '"' lw\_export-zport\_address1 '"' INTO lw\_einv\_api\_hdr-address1\_b.  
                          CONCATENATE '"' lw\_export-zport\_address2 '"' INTO lw\_einv\_api\_hdr-address2\_b.  
\*                          CONCATENATE '"' lw\_export-zport\_place '"' INTO lw\_einv\_api\_hdr-location\_b.  
                          lw\_einv\_api\_hdr-pincode\_b = lw\_export-zport\_pincode.  
  
                          CLEAR:lw\_t005u.  
                          READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_export-zport\_state.  
                          IF sy-subrc EQ 0.  
                            TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                            CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_einv\_api\_hdr-state\_code\_b.  
                          ENDIF.  
                          lw\_einv\_api\_hdr-place\_of\_supply\_b = gc\_export\_place.  
                          lw\_einv\_api\_hdr-state\_code\_b = gc\_exp\_state.  
                        ENDIF.  
                        lw\_einv\_api\_hdr-location\_b = gc\_export\_place."'"Other countries"'."Added by Lakshmi  
                      ENDIF.  
  
                    ENDIF.  
                  ENDIF.  
\*\*End of buyer\_details  
  
  
\*begin of dispatch\_details  
                  CONCATENATE '"' lw\_t001w-name1 '"' INTO lw\_einv\_api\_hdr-comapny\_name\_d.  
                  CONCATENATE '"' lw\_t001w-name2 '"' INTO   lw\_einv\_api\_hdr-address1\_d .  
                  CONCATENATE '"' lw\_t001w-stras '"' INTO  lw\_einv\_api\_hdr-address2\_d .  
                  CONCATENATE '"' lw\_t001w-ort01 '"'  INTO     lw\_einv\_api\_hdr-location\_d .  
                  IF lw\_t001w-pstlz IS  NOT INITIAL.  
                    lw\_einv\_api\_hdr-pincode\_d = lw\_t001w-pstlz .  
                  ENDIF.  
                  CLEAR:lw\_t005u\_d.  
                  READ TABLE gt\_t005u\_s INTO lw\_t005u\_s WITH  KEY bland = lw\_t001w-regio.  
                  IF sy-subrc EQ 0.  
                    TRANSLATE lw\_t005u\_s-bezei TO UPPER CASE.  
                    IF lw\_t005u\_s-bezei = 'ANDRA PRADESH'.  
                      lw\_t005u\_s-bezei = 'ANDHRA PRADESH'.  
                    ENDIF.  
\*                    if lw\_t005u\_s-bezei = 'UTTARANCHAL'.  
\*                      lw\_t005u\_s-bezei = 'UTTARAKHAND'.  
\*                    endif.  
                    CONCATENATE '"' lw\_t005u\_s-bezei '"'  INTO lw\_einv\_api\_hdr-state\_code\_d .  
                  ENDIF.  
  
\*begin of dispatch\_details.  
\*  
\*\*Begin of ship\_details  
  
                  CLEAR:lw\_vbpa.  
                  READ TABLE gt\_vbpa INTO lw\_vbpa WITH  KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                            parvw = gc\_parvw\_we  
                                                            BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    CLEAR:lw\_kna1\_d.  
                    READ TABLE gt\_kna1 INTO lw\_kna1\_d WITH  KEY kunnr = lw\_vbpa-kunnr BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"'  lw\_kna1\_d-stcd3 '"' INTO lw\_einv\_api\_hdr-gstin\_sh .  
                      CONCATENATE '"' lw\_kna1\_d-name1 '"' INTO lw\_einv\_api\_hdr-legal\_name\_sh.  
                      CONCATENATE '"' lw\_kna1\_d-name1 '"' INTO lw\_einv\_api\_hdr-trade\_name\_sh.  
\*                      CONCATENATE '"' lw\_kna1\_d-name2 '"' INTO   lw\_einv\_api\_hdr-address1\_sh .  
                      CONCATENATE '"' lw\_kna1\_d-name2 '"' INTO  lw\_einv\_api\_hdr-address2\_sh .  
                      IF lw\_kna1\_d-stras IS NOT INITIAL.  
                        lw\_einv\_api\_hdr-address1\_sh = lw\_kna1\_d-stras.  
                      ENDIF.  
                      IF lw\_kna1\_d-ort01  IS NOT INITIAL AND  lw\_einv\_api\_hdr-address1\_sh  IS INITIAL.  
                        lw\_einv\_api\_hdr-address1\_sh = lw\_kna1\_d-ort01.  
                      ENDIF.  
  
                      CONCATENATE '"' lw\_einv\_api\_hdr-address1\_sh '"' INTO lw\_einv\_api\_hdr-address1\_sh.  
                      IF lw\_kna1\_d-name2 IS NOT INITIAL AND lw\_kna1\_d-stras IS INITIAL.  
                        lw\_einv\_api\_hdr-address1\_sh = lw\_einv\_api\_hdr-address2\_sh .  
                      ENDIF.  
  
  
                      CONCATENATE '"'  lw\_kna1\_d-ort01 '"'  INTO     lw\_einv\_api\_hdr-location\_sh .  
                      IF lw\_kna1\_d-pstlz IS  NOT INITIAL.  
                        lw\_einv\_api\_hdr-pincode\_sh = lw\_kna1\_d-pstlz."lw\_einv\_api\_hdr-pincode\_b ."Changed by lakshmi  
                      ENDIF.  
                      CLEAR:lw\_t005u\_d.  
                      READ TABLE gt\_t005u INTO lw\_t005u\_d WITH  KEY bland = lw\_kna1\_d-regio.  
                      IF sy-subrc EQ 0.  
                        TRANSLATE lw\_t005u\_d-bezei TO UPPER CASE.  
                        IF lw\_t005u\_d-bezei = 'ANDRA PRADESH'.  
                          lw\_t005u\_d-bezei = 'ANDHRA PRADESH'.  
                        ENDIF.  
\*                        if lw\_t005u\_s-bezei = 'UTTARANCHAL'.  
\*                          lw\_t005u\_s-bezei = 'UTTARAKHAND'.  
\*                        endif.  
                        lw\_einv\_api\_hdr-state\_code\_sh  = lw\_t005u\_d-bezei."lw\_einv\_api\_hdr-state\_code\_b."Changed by lakshmi  
                      ENDIF.  
                      CONCATENATE '"' lw\_einv\_api\_hdr-state\_code\_sh '"' INTO lw\_einv\_api\_hdr-state\_code\_sh.  
                      CLEAR lv\_export.  
                      IF lw\_t001w-land1 NE lw\_kna1\_d-land1.  
                        lv\_export = abap\_true.  
                        lw\_einv\_api\_hdr-gstin\_sh = gc\_gstin\_urp.  
                        READ TABLE gt\_export INTO lw\_export WITH KEY zport\_code = gw\_final-zport\_code.  
                        IF sy-subrc IS INITIAL.  
                          CONCATENATE '"' lw\_export-zport\_address1 '"' INTO lw\_einv\_api\_hdr-address1\_sh.  
                          CONCATENATE '"' lw\_export-zport\_address2 '"' INTO lw\_einv\_api\_hdr-address2\_sh.  
                          CONCATENATE '"' lw\_export-zport\_place '"' INTO lw\_einv\_api\_hdr-location\_sh.  
                          lw\_einv\_api\_hdr-pincode\_sh = lw\_export-zport\_pincode.  
  
                          CLEAR:lw\_t005u\_d.  
                          READ TABLE gt\_t005u INTO lw\_t005u\_d WITH  KEY bland = lw\_export-zport\_state.  
                          IF sy-subrc EQ 0.  
                            TRANSLATE lw\_t005u\_d-bezei TO UPPER CASE.  
                            CONCATENATE '"' lw\_t005u\_d-bezei '"' INTO lw\_einv\_api\_hdr-state\_code\_sh.  
                          ENDIF.  
                        ENDIF.  
                      ENDIF.  
                    ENDIF.  
                  ENDIF.  
  
\*End of  ship\_details  
  
                  IF lw\_kna1-kunnr NE lw\_kna1\_d-kunnr .  
                    lw\_einv\_api\_hdr-transaction\_type = gc\_ttype\_shg.  
                  ELSEIF lw\_kna1-kunnr EQ lw\_kna1\_d-kunnr.  
                    lw\_einv\_api\_hdr-transaction\_type = gc\_ttype\_reg.  
                  ENDIF.  
  
\*Begin of export\_details  
                  IF lw\_t001w-land1 NE lw\_kna1-land1.  "Seller and Buyer country is different, then its export  
                    lw\_einv\_api\_hdr-supply\_type   = gc\_sup\_type\_expwp.  
                    lw\_einv\_api\_hdr-ship\_bill\_number = lw\_einv\_api\_hdr-document\_number.  
                    lw\_einv\_api\_hdr-ship\_bill\_date = lw\_einv\_api\_hdr-document\_date.  
                    CONCATENATE '"' lw\_kna1-land1             '"' INTO lw\_einv\_api\_hdr-country\_code.  
                    CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-waerk '"' INTO lw\_einv\_api\_hdr-foreign\_currency.  
                    CONCATENATE '"' ' '                       '"' INTO lw\_einv\_api\_hdr-refund\_claim.  
                    CONCATENATE '"' ' '                       '"' INTO lw\_einv\_api\_hdr-port\_code  .  
                  ELSE.  
                    CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-ship\_bill\_number.  
                    CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-ship\_bill\_date.  
                    CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-country\_code.  
                    CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-foreign\_currency.  
                    CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-refund\_claim.  
                    CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-port\_code  .  
                  ENDIF.  
\*End of export\_details  
  
\*\*Begin of payment\_details  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-bank\_account\_number.  
                  CONCATENATE ' ' '0' ' ' INTO lw\_einv\_api\_hdr-paid\_balance\_amount.  
                  CONCATENATE ' ' '0' ' ' INTO lw\_einv\_api\_hdr-credit\_days.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-credit\_transfer.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-direct\_debit.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-branch\_or\_ifsc.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_mode.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payee\_name.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_due\_date.  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_instruction .  
                  CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_term.  
  
\*\*End of payment\_details  
  
  
                  lv\_tdname  = gw\_final-vbeln.  
  
                  CALL FUNCTION 'READ\_TEXT'  
                    EXPORTING  
                      client                  = sy-mandt  
                      id                      = gc\_tdid\_0002  
                      language                = sy-langu  
                      name                    = lv\_tdname  
                      object                  = gc\_tobject\_vbbk  
                    TABLES  
                      lines                   = lt\_lines  
                    EXCEPTIONS  
                      id                      = 1  
                      language                = 2  
                      name                    = 3  
                      not\_found               = 4  
                      object                  = 5  
                      reference\_check         = 6  
                      wrong\_access\_to\_archive = 7  
                      OTHERS                  = 8.  
                  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
                    CONCATENATE  '"' '' '"' INTO lw\_einv\_api\_hdr-invoice\_remarks. "Required  
                  ELSE.  
                    READ TABLE lt\_lines INTO lw\_lines INDEX 1.  
                    CONCATENATE  '"' lw\_lines-tdline '"' INTO lw\_einv\_api\_hdr-invoice\_remarks. "Required  
  
                  ENDIF.  
  
\*Begin of reference\_details  
                  lw\_einv\_api\_hdr-invoice\_period\_start\_date = lw\_einv\_api\_hdr-document\_date.  
                  lw\_einv\_api\_hdr-invoice\_period\_end\_date   = lw\_einv\_api\_hdr-document\_date.  
  
\*\*Begin of preceding\_document\_details  
                  lw\_einv\_api\_hdr-reference\_of\_original\_invoice = lw\_einv\_api\_hdr-document\_number.  
                  lw\_einv\_api\_hdr-preceding\_invoice\_date        = lw\_einv\_api\_hdr-document\_date.  
                  lw\_einv\_api\_hdr-other\_reference = '""'.  
\*End of preceding\_document\_details  
\*End of reference\_details  
  
\*Begin of contract Details,  
                  CLEAR: lw\_vbfa, lw\_wb2\_v\_vbak\_vbap2.  
                  READ TABLE gt\_vbfa INTO lw\_vbfa WITH KEY vbeln = gw\_final-vbeln.  
                  IF sy-subrc EQ 0.  
                    READ TABLE gt\_wb2\_v\_vbak\_vbap2 INTO lw\_wb2\_v\_vbak\_vbap2 WITH KEY vbeln = lw\_vbfa-vbelv  
                                                                                     posnr\_i = lw\_vbfa-posnv  
                                                                                     BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"' lw\_wb2\_v\_vbak\_vbap2-vbeln  '"' INTO    lw\_einv\_api\_hdr-receipt\_advice\_number .  
                      IF lw\_wb2\_v\_vbak\_vbap2-audat IS NOT INITIAL.  
                        CONCATENATE '"' lw\_wb2\_v\_vbak\_vbap2-audat+6(2) '/'  
                                        lw\_wb2\_v\_vbak\_vbap2-audat+4(2) '/'  
                                        lw\_wb2\_v\_vbak\_vbap2-audat+0(4)  
                                    '"' INTO    lw\_einv\_api\_hdr-receipt\_advice\_date .  
                      ENDIF.  
                      CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-charg\_i '"' INTO    lw\_einv\_api\_hdr-batch\_reference\_number.  
                      CONCATENATE '"' lw\_wb2\_v\_vbak\_vbap2-bname '"' INTO    lw\_einv\_api\_hdr-contract\_reference\_number.  
                      CONCATENATE '"' lw\_wb2\_v\_vbak\_vbap2-ihrez '"' INTO    lw\_einv\_api\_hdr-other\_reference\_c.  
                      lw\_einv\_api\_hdr-project\_reference\_number = '""'.  
                      CONCATENATE '"' lw\_wb2\_v\_vbak\_vbap2-bstnk '"' INTO    lw\_einv\_api\_hdr-vendor\_po\_reference\_number.  
  
                      IF lw\_wb2\_v\_vbak\_vbap2-bstdk IS NOT INITIAL.  
                        CONCATENATE '"' lw\_wb2\_v\_vbak\_vbap2-bstdk+6(2) '/'  
                                        lw\_wb2\_v\_vbak\_vbap2-bstdk+4(2) '/'  
                                        lw\_wb2\_v\_vbak\_vbap2-bstdk+0(4)  
                                    '"' INTO    lw\_einv\_api\_hdr-vendor\_po\_reference\_date.  
                      ELSE.  
                        lw\_einv\_api\_hdr-vendor\_po\_reference\_date = '""'.  
                      ENDIF.  
                      "End of contract\_details  
                    ENDIF.  
                  ENDIF.  
\*Begin of'"additional\_document\_details"  
                  lw\_einv\_api\_hdr-supporting\_document\_url = '""'.  
                  lw\_einv\_api\_hdr-supporting\_document     = '""'.  
                  lw\_einv\_api\_hdr-additional\_information  = '""'.  
  
  
\*End of'"additional\_document\_details"  
  
\*Begin of value\_details  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
                  lw\_einv\_api\_hdr-total\_assessable\_value          = '0'.  
                  lw\_einv\_api\_hdr-total\_cgst\_value                = '0'.  
                  lw\_einv\_api\_hdr-total\_sgst\_value                = '0'.  
                  lw\_einv\_api\_hdr-total\_igst\_value                = '0'.  
                  lw\_einv\_api\_hdr-total\_cess\_value                = '0'.  
                  lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value       = '0'.  
                  lw\_einv\_api\_hdr-total\_invoice\_value             = '0'.  
                  lw\_einv\_api\_hdr-total\_cess\_value\_of\_state       = '0'.  
                  lw\_einv\_api\_hdr-round\_off\_amount                = '0'.  
                  lw\_einv\_api\_hdr-total\_invoice\_value\_additional  = '0'.  
                  lw\_einv\_api\_hdr-total\_other\_charge              = '0'.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
                  lw\_einv\_api\_hdr-total\_assessable\_value = '1'.               "Required  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*Begin of item\_list  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
                  lw\_einv\_api\_itm-quantity                    = '0'.  
                  lw\_einv\_api\_itm-free\_quantity               = '0'.  
                  lw\_einv\_api\_itm-unit\_price                  = '0'.  
                  lw\_einv\_api\_itm-total\_amount                = '0'.  
                  lw\_einv\_api\_itm-pre\_tax\_value               = '0'.  
                  lw\_einv\_api\_itm-discount                    = '0'.  
                  lw\_einv\_api\_itm-other\_charge                = '0'.  
                  lw\_einv\_api\_itm-gst\_rate                    = '0'.  
                  lw\_einv\_api\_itm-igst\_amount                 = '0'.  
                  lw\_einv\_api\_itm-cgst\_amount                 = '0'.  
                  lw\_einv\_api\_itm-sgst\_amount                 = '0'.  
                  lw\_einv\_api\_itm-cess\_rate                   = '0'.  
                  lw\_einv\_api\_itm-cess\_amount                 = '0'.  
                  lw\_einv\_api\_itm-cess\_nonadvol\_value         = '0'.  
                  lw\_einv\_api\_itm-state\_cess\_rate             = '0'.  
                  lw\_einv\_api\_itm-state\_cess\_amount           = '0'.  
                  lw\_einv\_api\_itm-state\_cess\_nonadvol\_amount  = '0'.  
                  lw\_einv\_api\_itm-total\_item\_value            = '0'.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\* check for export currency  
                  CLEAR lv\_exp\_curr.  
                  IF lw\_wb2\_v\_vbrk\_vbrp2-waerk NE  gc\_curr\_inr.  
                    lv\_exp\_curr = abap\_true.  
                  ENDIF.  
  
  
                  lw\_einv\_api\_itm-bukrs  = lw\_wb2\_v\_vbrk\_vbrp2-bukrs.  
                  lw\_einv\_api\_itm-doc\_year = gw\_final-gjahr.  
                  lw\_einv\_api\_itm-doc\_type = lw\_wb2\_v\_vbrk\_vbrp2-fkart.  
                  lw\_einv\_api\_itm-document\_type = lw\_wb2\_v\_vbrk\_vbrp2-fkart  .  
                  lw\_einv\_api\_itm-document\_number = lw\_einv\_api\_hdr-document\_number   .  
                  SHIFT lw\_einv\_api\_itm-document\_number LEFT DELETING LEADING '0'.  
                  IF lw\_wb2\_v\_vbrk\_vbrp2-fkdat IS NOT INITIAL.  
                    CONCATENATE  lw\_wb2\_v\_vbrk\_vbrp2-fkdat+6(2) '/'  
                                 lw\_wb2\_v\_vbrk\_vbrp2-fkdat+4(2) '/'  
                                 lw\_wb2\_v\_vbrk\_vbrp2-fkdat+0(4)  
                            INTO lw\_einv\_api\_itm-document\_date.  
                  ENDIF.  
                  CONCATENATE '"' lw\_einv\_api\_itm-document\_type '"' INTO  lw\_einv\_api\_itm-document\_type.  
                  CONCATENATE '"' lw\_einv\_api\_itm-document\_date '"' INTO  lw\_einv\_api\_itm-document\_date.  
  
  
  
                  CONCATENATE '"'  lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i '"' INTO lw\_einv\_api\_itm-item\_serial\_number .  
                  CONCATENATE '"'  lw\_wb2\_v\_vbrk\_vbrp2-arktx\_i '"' INTO lw\_einv\_api\_itm-product\_description .  
  
                  IF lw\_wb2\_v\_vbrk\_vbrp2-pstyv\_i EQ gc\_pstyv\_tad.  
                    lw\_einv\_api\_itm-is\_service = gc\_service\_y.  
                  ELSE.  
                    lw\_einv\_api\_itm-is\_service = gc\_service\_n.  
                  ENDIF.  
  
                  READ TABLE gt\_marc INTO lw\_marc WITH KEY matnr = lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i  
                                                           werks = lw\_wb2\_v\_vbrk\_vbrp2-werks\_i  
                                                           BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    CONCATENATE '"'  lw\_marc-steuc  '"' INTO lw\_einv\_api\_itm-hsn\_code.  
                  ENDIF.  
                  lw\_einv\_api\_itm-bar\_code =  '""'.  
                  "Quantity  
                  lw\_einv\_api\_itm-quantity = lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i . "'1'.  
                  lw\_einv\_api\_itm-free\_quantity = '0'.  
  
\*Unit  
                  CLEAR lv\_output\_uom.  
                  CALL FUNCTION 'CONVERSION\_EXIT\_CUNIT\_OUTPUT'  
                    EXPORTING  
                      input          = lw\_wb2\_v\_vbrk\_vbrp2-meins\_i  
                    IMPORTING  
                      output         = lv\_output\_uom  
                    EXCEPTIONS  
                      unit\_not\_found = 1  
                      OTHERS         = 2.  
                  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
                  ENDIF.  
  
                  READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins = lv\_output\_uom.  
                  IF sy-subrc EQ 0.  
                    CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_einv\_api\_itm-unit.  
                  ELSE.  
                    READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins =  lw\_wb2\_v\_vbrk\_vbrp2-meins\_i.  
                    IF sy-subrc IS INITIAL.  
                      CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_einv\_api\_itm-unit.  
                    ELSE.  
                      CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-meins\_i '"' INTO  lw\_einv\_api\_itm-unit.  
                    ENDIF.  
                  ENDIF.  
  
\* Unit Price  
                  IF lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i IS NOT INITIAL  .  
                    CLEAR:lv\_rate,lv\_dmbtr,lv\_item\_amt.  
\*\*                    LOOP AT gt\_tvarvc INTO lw\_tvarvc WHERE name = gc\_var\_unit.  
\*\*                      READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
\*\*                                                                                 kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
\*\*                                                                                 kschl = lw\_tvarvc-low.  
\*\*                      IF sy-subrc IS INITIAL.  
\*\*                        EXIT.  
\*\*                      ENDIF.  
\*\*                    ENDLOOP.  
\*\*                    IF lw\_konv IS NOT INITIAL.  
\*\*                      CLEAR:lv\_rate,lv\_dmbtr,lv\_item\_amt.  
\*\*                      IF lv\_exp\_curr IS NOT INITIAL.  
\*\*\*                        READ TABLE gt\_bkpfm INTO lw\_bkpfm WITH KEY bukrs = lw\_wb2\_v\_vbrk\_vbrp2-bukrs  
\*\*\*                                                                    xblnr = lw\_wb2\_v\_vbrk\_vbrp2-xblnr.  
\*\*\*                        IF sy-subrc IS INITIAL.  
\*\*\*                          READ TABLE gt\_bseg INTO lw\_bseg WITH KEY bukrs = lw\_bkpfm-bukrs  
\*\*\*                                                                   belnr = lw\_bkpfm-belnr  
\*\*\*                                                                   gjahr = lw\_bkpfm-gjahr  
\*\*\*                                                                   txgrp = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i.  
\*\*\*                        ENDIF.  
\*\*  
\*\*                        lv\_rate = lw\_konv-kbetr \* lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i.  
\*\*                        lw\_einv\_api\_itm-unit\_price = lv\_rate.  
\*\*                        lv\_dmbtr = lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i \* lv\_rate.  
\*\*                        lv\_item\_amt = ( ( lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i + lw\_wb2\_v\_vbrk\_vbrp2-mwsbp\_i ) \*  
\*\*                                       lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i )."  "lw\_bseg-dmbtr.  
\*\*                      ELSE.  
\*\*  
\*\*                        lv\_rate = lw\_konv-kbetr.  
\*\*                        lw\_einv\_api\_itm-unit\_price = lv\_rate.  
\*\*                        lv\_dmbtr = lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i \* lv\_rate.  
\*\*                        lw\_einv\_api\_itm-total\_amount = lv\_dmbtr.  
\*\*                        lv\_item\_amt = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i + lw\_wb2\_v\_vbrk\_vbrp2-mwsbp\_i.  
\*\*                      ENDIF.  
\*\*                    ENDIF.  
                    CLEAR lw\_konv.  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                              kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                              kschl = 'DIFF'  
                                                              BINARY SEARCH.  
  
                    IF lv\_exp\_curr IS NOT INITIAL.  
  
                      lv\_rate = ( lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i - lw\_konv-kwert  ) / lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i.  
                      lw\_einv\_api\_itm-unit\_price = lv\_rate.  
  
                      PERFORM convert\_currency USING lw\_einv\_api\_itm-unit\_price lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING lw\_einv\_api\_itm-unit\_price.  
  
                      lw\_einv\_api\_itm-total\_amount = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i.  
  
                      PERFORM convert\_currency USING lw\_einv\_api\_itm-total\_amount lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING lw\_einv\_api\_itm-total\_amount.  
  
                      lv\_item\_amt = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i + lw\_wb2\_v\_vbrk\_vbrp2-mwsbp\_i.  
  
                      PERFORM convert\_currency USING lv\_item\_amt lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                 CHANGING lv\_item\_amt.  
  
  
                    ELSE.  
                      lv\_rate = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i / lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i.  
                      lw\_einv\_api\_itm-unit\_price = lv\_rate.  
                      lw\_einv\_api\_itm-total\_amount = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i.  
                      lv\_item\_amt = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i + lw\_wb2\_v\_vbrk\_vbrp2-mwsbp\_i.  
                    ENDIF.  
  
                    CONDENSE:lw\_einv\_api\_itm-unit\_price,lw\_einv\_api\_itm-total\_amount.  
                  ENDIF.  
                  lw\_einv\_api\_itm-pre\_tax\_value =  '0'.  
                  CONDENSE:lw\_einv\_api\_itm-pre\_tax\_value.  
  
                  "Discount amount  
                  LOOP AT gt\_tvarvc INTO lw\_tvarvc WHERE name = gc\_var\_disc.  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = lw\_tvarvc-low  
                                                             BINARY SEARCH.  
  
                    IF sy-subrc EQ 0.  
                      lw\_konv-kwert = lw\_konv-kwert \* -1.  
                      IF lv\_exp\_curr IS NOT INITIAL.  
                        lw\_konv-kwert = lw\_konv-kwert \* lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i.  
                      ENDIF.  
  
                      lw\_einv\_api\_itm-discount =  lw\_einv\_api\_itm-discount + lw\_konv-kwert.  
                    ENDIF.  
                  ENDLOOP.  
                  IF lw\_einv\_api\_itm-discount IS NOT INITIAL AND lv\_exp\_curr IS NOT INITIAL.  
                    PERFORM convert\_currency USING lw\_einv\_api\_itm-discount lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                               CHANGING lw\_einv\_api\_itm-discount.  
                  ENDIF.  
                  CONDENSE:lw\_einv\_api\_itm-discount.  
  
                  lw\_einv\_api\_itm-other\_charge =  '0'.  
                  lw\_einv\_api\_itm-assessable\_value = lw\_einv\_api\_itm-total\_amount -  
                                                     lw\_einv\_api\_itm-discount. "Required  
                  CONDENSE:lw\_einv\_api\_itm-assessable\_value,lw\_einv\_api\_itm-other\_charge.  
                  lv\_total\_assessable\_value = lv\_total\_assessable\_value  + lw\_einv\_api\_itm-assessable\_value.  
                  "Other Charges  
\*                  Code begin added by Lakshmi  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_jwts  
                                                             BINARY SEARCH.  
                  IF sy-subrc = 0.  
                    lw\_einv\_api\_itm-other\_charge = lw\_einv\_api\_itm-other\_charge + lw\_konv-kwert.  
                  ENDIF.  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_jtc1  
                                                             BINARY SEARCH.  
                  IF sy-subrc = 0.  
                    lw\_einv\_api\_itm-other\_charge = lw\_einv\_api\_itm-other\_charge + lw\_konv-kwert.  
                  ENDIF.  
                  CONDENSE lw\_einv\_api\_itm-other\_charge.  
\*                  Code ended added by Lakshmi  
                  CLEAR:lw\_gstrate.  
                  "IGST Tax amount  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                           kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                           kschl = gc\_kschl\_joig  
                                                           BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    IF lv\_exp\_curr IS NOT INITIAL.  
                      PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                             CHANGING lw\_konv-kwert.  
                    ENDIF.  
  
                    lw\_gstrate =  lw\_konv-kbetr . "Required  
                    lw\_einv\_api\_itm-igst\_amount =  lw\_konv-kwert.  
                    CONDENSE:lw\_einv\_api\_itm-gst\_rate,lw\_einv\_api\_itm-igst\_amount.  
                    lv\_total\_igst\_value = lv\_total\_igst\_value  + lw\_einv\_api\_itm-igst\_amount.  
                  ENDIF.  
  
                  "CGST Tax amount  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                           kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                           kschl = gc\_kschl\_jocg  
                                                           BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    IF lv\_exp\_curr IS NOT INITIAL.  
                      PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                             CHANGING lw\_konv-kwert.  
                    ENDIF.  
                    lw\_einv\_api\_itm-gst\_rate =  lw\_konv-kbetr. "Required  
                    CONDENSE lw\_einv\_api\_itm-gst\_rate.  
                    ADD lw\_einv\_api\_itm-gst\_rate TO lw\_gstrate.  
                    lw\_einv\_api\_itm-cgst\_amount =  lw\_konv-kwert.  
                    CONDENSE:lw\_einv\_api\_itm-gst\_rate ,lw\_einv\_api\_itm-cgst\_amount.  
                    lv\_total\_cgst\_value = lv\_total\_cgst\_value  + lw\_einv\_api\_itm-cgst\_amount.  
                  ENDIF.  
  
                  "SGST Tax amount  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                           kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                           kschl = gc\_kschl\_josg  
                                                           BINARY SEARCH.  
                  IF sy-subrc IS NOT INITIAL.  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_joug  
                                                             BINARY SEARCH.  
                  ENDIF.  
                  IF sy-subrc EQ 0.  
                    IF lv\_exp\_curr IS NOT INITIAL.  
                      PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                             CHANGING lw\_konv-kwert.  
                    ENDIF.  
  
                    lw\_einv\_api\_itm-gst\_rate =  lw\_konv-kbetr. "Required  
                    CONDENSE lw\_einv\_api\_itm-gst\_rate.  
                    ADD lw\_einv\_api\_itm-gst\_rate TO lw\_gstrate.  
                    lw\_einv\_api\_itm-sgst\_amount =  lw\_konv-kwert.  
                    CONDENSE:lw\_einv\_api\_itm-gst\_rate,lw\_einv\_api\_itm-sgst\_amount.  
                    lv\_total\_sgst\_value = lv\_total\_sgst\_value  + lw\_einv\_api\_itm-sgst\_amount.  
                  ENDIF.  
  
                  "CESS  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                           kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                           kschl = gc\_kschl\_cess  
                                                           BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    IF lv\_exp\_curr IS NOT INITIAL.  
                      PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                             CHANGING lw\_konv-kwert.  
                    ENDIF.  
                    lw\_einv\_api\_itm-cess\_rate =  lw\_konv-kbetr. "Required  
                    lw\_einv\_api\_itm-cess\_amount =  lw\_konv-kwert.  
                    CONDENSE:lw\_einv\_api\_itm-cess\_rate,lw\_einv\_api\_itm-cess\_amount.  
                    lv\_total\_cess\_value = lv\_total\_cess\_value  + lw\_einv\_api\_itm-cess\_amount.  
                  ENDIF.  
  
                  "cess\_nonadvol\_amount  
                  READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                           kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                           kschl = gc\_kschl\_cess  
                                                           BINARY SEARCH.  
                  IF sy-subrc EQ 0.  
                    IF lv\_exp\_curr IS NOT INITIAL.  
                      PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                             CHANGING lw\_konv-kwert.  
                    ENDIF.  
                    lw\_einv\_api\_itm-cess\_nonadvol\_value =  lw\_konv-kwert.  
                    CONDENSE:lw\_einv\_api\_itm-cess\_nonadvol\_value.  
                    lv\_total\_cess\_nonadvol\_value = lv\_total\_cess\_nonadvol\_value  + lw\_einv\_api\_itm-cess\_nonadvol\_value.  
                  ENDIF.  
                  lw\_einv\_api\_itm-total\_item\_value =  ( lw\_einv\_api\_itm-assessable\_value +  
                                                        lw\_einv\_api\_itm-cgst\_amount +  
                                                        lw\_einv\_api\_itm-sgst\_amount +  
                                                        "lw\_einv\_api\_itm-cess\_amount +  
                                                        lw\_einv\_api\_itm-igst\_amount +  
                                                        lw\_einv\_api\_itm-state\_cess\_amount +  
                                                        lw\_einv\_api\_itm-cess\_nonadvol\_value +  
                                                        lw\_einv\_api\_itm-other\_charge ).  
  
                  CONDENSE:lw\_einv\_api\_itm-total\_item\_value.  
  
\* round off value  
\*                  CLEAR lv\_roundoff.  
\*                  IF lv\_item\_amt NE lw\_einv\_api\_itm-total\_item\_value.  
\*                    lv\_roundoff = lv\_item\_amt - lw\_einv\_api\_itm-total\_item\_value.  
\*                  ENDIF.  
  
  
                  lv\_total\_invoice\_value = lv\_total\_invoice\_value +  
                                           lw\_einv\_api\_itm-total\_item\_value.  
                  "lv\_roundoff.  
  
  
                  CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-land1 '"' INTO lw\_einv\_api\_itm-country\_origin."'"52"'.  
                  "get billing item text object VBBP  
                  CALL FUNCTION 'CONVERSION\_EXIT\_MATN1\_OUTPUT'  
                    EXPORTING  
                      input  = lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i  
                    IMPORTING  
                      output = lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i.  
  
                  CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i '"' INTO lw\_einv\_api\_itm-order\_line\_reference.  
                  CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i '"' INTO lw\_einv\_api\_itm-product\_serial\_number.  
  
\*             '"batch details":' '{'  
                  CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-charg\_i '"' INTO lw\_einv\_api\_itm-name." =   '"aaa"'.  "Required  
                  lw\_einv\_api\_itm-expiry\_date =  '" "'.  
                  lw\_einv\_api\_itm-warranty\_date = '" "'.  
  
\*Begin of  attribute\_details"  
                  lw\_einv\_api\_itm-item\_attribute\_details = gc\_attr\_einv.  
                  lw\_einv\_api\_itm-item\_attribute\_value = gc\_attr\_einv.  
\*End of of  attribute\_details"  
\*End of item\_list  
  
                  lw\_einv\_api\_itm-gst\_rate = lw\_gstrate.  
                  CONDENSE:lw\_einv\_api\_itm-gst\_rate.  
                  APPEND lw\_einv\_api\_itm TO gt\_einv\_api\_itm.  
                  CLEAR:lw\_einv\_api\_itm .  
                ENDLOOP.  
  
  
  
                lw\_einv\_api\_hdr-total\_assessable\_value = lv\_total\_assessable\_value.  
                lw\_einv\_api\_hdr-total\_cgst\_value =       lv\_total\_cgst\_value.  
                lw\_einv\_api\_hdr-total\_sgst\_value =  lv\_total\_sgst\_value.  
                lw\_einv\_api\_hdr-total\_igst\_value =   lv\_total\_igst\_value .  
                lw\_einv\_api\_hdr-total\_invoice\_value =  lv\_total\_invoice\_value.  
                lw\_einv\_api\_hdr-total\_cess\_value =  lv\_total\_cess\_value.  
                lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value =  lv\_total\_cess\_nonadvol\_value.  
                CONDENSE:lw\_einv\_api\_hdr-total\_assessable\_value,lw\_einv\_api\_hdr-total\_cgst\_value,lw\_einv\_api\_hdr-total\_sgst\_value,lw\_einv\_api\_hdr-total\_igst\_value,  
                         lw\_einv\_api\_hdr-total\_invoice\_value,lw\_einv\_api\_hdr-total\_cess\_value,lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value.  
                "Added by Lakshmi  
                IF lw\_t001w-land1 NE lw\_kna1-land1.  "Seller and Buyer country is different, then its export  
                  IF lw\_einv\_api\_hdr-total\_igst\_value = '0.00'." is INITIAL.  
                    lw\_einv\_api\_hdr-supply\_type   = gc\_sup\_type\_expwop.  
                  ELSE.  
                    lw\_einv\_api\_hdr-supply\_type = gc\_sup\_type\_expwp.  
                  ENDIF.  
                ENDIF.  
                "Added by Lakshmi  
                APPEND lw\_einv\_api\_hdr TO gt\_einv\_api\_hdr.  
                CLEAR:lw\_einv\_api\_hdr.  
  
              ENDIF.  
            ENDIF.  
          ELSEIF p\_mod = gc\_fi.  
  
            SORT gt\_bseg BY belnr buzei.  
            READ TABLE gt\_bseg INTO gw\_bseg WITH KEY belnr = gw\_final-vbeln.  
            IF sy-subrc = 0.  
              lv\_index = sy-tabix.  
              lv\_total\_assessable\_value    = 0.  
              lv\_total\_cgst\_value          = 0.  
              lv\_total\_sgst\_value          = 0.  
              lv\_total\_igst\_value          = 0.  
              lv\_total\_invoice\_value       = 0.  
              lv\_total\_cess\_value          = 0.  
              lv\_total\_cess\_nonadvol\_value = 0.  
              CLEAR:lw\_einv\_api\_hdr-total\_assessable\_value,lw\_einv\_api\_hdr-total\_cgst\_value,lw\_einv\_api\_hdr-total\_sgst\_value,lw\_einv\_api\_hdr-total\_igst\_value,  
                                   lw\_einv\_api\_hdr-total\_invoice\_value,lw\_einv\_api\_hdr-total\_cess\_value,lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value.  
  
              CLEAR: lv\_total\_assessable\_value,lv\_total\_igst\_value,  lv\_total\_cgst\_value,lv\_total\_sgst\_value, lv\_total\_cess\_value,lv\_total\_cess\_nonadvol\_value, lv\_total\_invoice\_value.  
  
              READ TABLE gt\_bkpf INTO gw\_bkpf WITH KEY bukrs = gw\_bseg-bukrs  
                                                             belnr = gw\_bseg-belnr  
                                                             gjahr = gw\_bseg-gjahr.  
              IF sy-subrc IS INITIAL.  
                lw\_einv\_api\_hdr-vbeln  = gw\_bseg-belnr.  
                lw\_einv\_api\_hdr-bukrs  = gw\_bseg-bukrs.  
                lw\_einv\_api\_hdr-doc\_year = gw\_bseg-gjahr.  
                lw\_einv\_api\_hdr-doc\_type = gw\_bkpf-blart.  
  
\*Data to send to the API IN JSON Scheme  
                READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_usrgstin.  
                IF sy-subrc IS INITIAL.  
                  lw\_einv\_api\_hdr-user\_gstin   = lw\_usrgstin-apiprov.  
                ELSE.  
                  lw\_einv\_api\_hdr-user\_gstin   =  gw\_final-sup\_gstin.  
                ENDIF.  
                CONCATENATE  '"' lw\_einv\_api\_hdr-user\_gstin '"' INTO lw\_einv\_api\_hdr-user\_gstin.  
  
                lw\_einv\_api\_hdr-supply\_type   = gc\_sup\_type\_b2b.  
  
                lw\_einv\_api\_hdr-charge\_type = gc\_charge\_n.  
                lw\_einv\_api\_hdr-ecommerce\_gstin = '""'.  
  
                lw\_einv\_api\_hdr-document\_number = gw\_bkpf-xblnr. "Changed by Lakshmi  
                SHIFT lw\_einv\_api\_hdr-document\_number LEFT DELETING LEADING '0'.  
                IF gw\_bkpf-budat IS NOT INITIAL.  
                  CONCATENATE  gw\_bkpf-budat+6(2) '/'  
                               gw\_bkpf-budat+4(2) '/'  
                               gw\_bkpf-budat+0(4)  
                          INTO lw\_einv\_api\_hdr-document\_date.  
                ENDIF.  
  
                READ TABLE gt\_doctyp INTO lw\_doctyp WITH KEY  bukrs = gw\_bkpf-bukrs  
                                                              fkart = gw\_bkpf-blart  
                                                              zmodule = gc\_fi.  
                IF sy-subrc = 0.  
                  lw\_einv\_api\_hdr-document\_type = lw\_doctyp-edoc\_type.  
                ELSE.  
                  lw\_einv\_api\_hdr-document\_type = gc\_docty\_inv.  
                ENDIF.  
                CONCATENATE '"' lw\_einv\_api\_hdr-document\_type '"' INTO  lw\_einv\_api\_hdr-document\_type.  
                CONCATENATE '"' lw\_einv\_api\_hdr-document\_number '"' INTO  lw\_einv\_api\_hdr-document\_number.  
                CONCATENATE '"' lw\_einv\_api\_hdr-document\_date '"' INTO  lw\_einv\_api\_hdr-document\_date.  
  
\*\*Begin of seller\_details  
                CLEAR: lw\_gstin, lw\_adrc.  
                READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_usrgstin.  
                IF sy-subrc IS INITIAL.  
                  lw\_einv\_api\_hdr-gstin\_s   = lw\_usrgstin-apiprov.  
                ELSE.  
                  lw\_einv\_api\_hdr-gstin\_s   =  gw\_final-sup\_gstin.  
                ENDIF.  
\*                lw\_einv\_api\_hdr-gstin\_s   =  gw\_final-sup\_gstin.  
                CONCATENATE '"' lw\_einv\_api\_hdr-gstin\_s '"' INTO lw\_einv\_api\_hdr-gstin\_s.  
  
                CLEAR lw\_gstin.  
                READ TABLE gt\_gstin INTO lw\_gstin WITH KEY bukrs = gw\_bseg-bukrs  
                                                          branch = gw\_bseg-bupla.  
\*                IF sy-subrc EQ 0.  
                CLEAR lw\_adrc1.  
                READ TABLE gt\_adrc1 INTO lw\_adrc1 WITH  KEY addrnumber = lw\_gstin-adrnr.  
\*                  IF sy-subrc IS INITIAL.  
                CONCATENATE '"'   lw\_adrc1-name1  '"' INTO   lw\_einv\_api\_hdr-legal\_name\_s.  
                CONCATENATE '"'   lw\_adrc1-name1  '"' INTO   lw\_einv\_api\_hdr-trade\_name\_s.  
                IF lw\_adrc1-name2 IS NOT INITIAL.  
                  CONCATENATE '"'   lw\_adrc1-name2  '"' INTO   lw\_einv\_api\_hdr-address1\_s .  
                ELSE.  
  
                  CONCATENATE '"'   lw\_adrc1-name1  '"' INTO   lw\_einv\_api\_hdr-address1\_s .  
  
                ENDIF.  
                CONCATENATE '"'   lw\_adrc1-street  '"' INTO   lw\_einv\_api\_hdr-address2\_s.  
                CONCATENATE '"'   lw\_adrc1-city1  '"' INTO   lw\_einv\_api\_hdr-location\_s.  
  
                lw\_einv\_api\_hdr-pincode\_s =  lw\_adrc1-post\_code1   .  
  
                READ TABLE gt\_adr61 INTO lw\_adr61 WITH KEY addrnumber = lw\_adrc1-addrnumber.  
                CONCATENATE '"' lw\_adr61-smtp\_addr '"'INTO lw\_einv\_api\_hdr-email\_s.  
                CLEAR lw\_t005u.  
                READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_adrc1-region.  
\*                    IF sy-subrc EQ 0.  
                TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                CONCATENATE '"'   lw\_t005u-bezei  '"' INTO  lw\_einv\_api\_hdr-state\_code\_s .  
\*                    ENDIF.  
  
\*                  ENDIF.  
\*                ENDIF.  
              ENDIF.  
\*\*End of seller\_details  
  
\*\*Begin of buyer\_details  
              IF gw\_bseg-kunnr IS NOT INITIAL.  
                CLEAR lw\_kna1.  
                READ TABLE gt\_kna1 INTO lw\_kna1 WITH  KEY kunnr = gw\_bseg-kunnr.  
\*                IF sy-subrc IS INITIAL.  
                CONCATENATE '"'  lw\_kna1-stcd3 '"' INTO          lw\_einv\_api\_hdr-gstin\_b .  
                CONCATENATE '"'  lw\_kna1-name1 '"' INTO          lw\_einv\_api\_hdr-legal\_name\_b.  
                CONCATENATE '"' lw\_kna1-name1 '"' INTO lw\_einv\_api\_hdr-trade\_name\_b  .  
                IF lw\_kna1-name2 IS NOT INITIAL.  
                  CONCATENATE '"' lw\_kna1-name2 '"' INTO          lw\_einv\_api\_hdr-address1\_b.  
                ELSE.  
                  CONCATENATE '"' lw\_kna1-name1 '"' INTO          lw\_einv\_api\_hdr-address1\_b.  
                ENDIF.  
                CONCATENATE '"' lw\_kna1-stras '"' INTO  lw\_einv\_api\_hdr-address2\_b.  
                CONCATENATE '"'  lw\_kna1-ort01 '"'  INTO           lw\_einv\_api\_hdr-location\_b.  
                lw\_einv\_api\_hdr-pincode\_b  = lw\_kna1-pstlz.  
  
                lw\_einv\_api\_hdr-place\_of\_supply\_b = lw\_kna1-regio.  
  
                CONCATENATE '"' lw\_einv\_api\_hdr-place\_of\_supply\_b '"' INTO lw\_einv\_api\_hdr-place\_of\_supply\_b.  
                CLEAR lw\_t005u.  
                READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_kna1-regio.  
\*                  IF sy-subrc EQ 0.  
                TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                CONCATENATE '"' lw\_t005u-bezei '"'  INTO  lw\_einv\_api\_hdr-state\_code\_b.  
                lw\_einv\_api\_hdr-place\_of\_supply\_b   =   lw\_einv\_api\_hdr-state\_code\_b.                                           "State code    Y  
\*                  ENDIF.  
\*                ENDIF.  
  
  
              ELSEIF  gw\_bseg-lifnr IS NOT INITIAL.  
                CLEAR lw\_lfa1.  
                READ TABLE gt\_lfa1 INTO lw\_lfa1 WITH KEY lifnr = gw\_bseg-lifnr.  
  
\*                IF sy-subrc IS INITIAL.  
                CONCATENATE '"'   lw\_lfa1-stcd3 '"' INTO          lw\_einv\_api\_hdr-gstin\_b .  
                CONCATENATE '"'  lw\_lfa1-name1 '"' INTO          lw\_einv\_api\_hdr-legal\_name\_b.  
                CONCATENATE '"' lw\_lfa1-name1 '"' INTO lw\_einv\_api\_hdr-trade\_name\_b  .  
                IF lw\_lfa1-name2 IS NOT INITIAL.  
                  CONCATENATE '"' lw\_lfa1-name2 '"' INTO          lw\_einv\_api\_hdr-address1\_b.  
                ELSE.  
                  CONCATENATE '"' lw\_lfa1-name1 '"' INTO          lw\_einv\_api\_hdr-address1\_b.  
                ENDIF.  
                CONCATENATE '"' lw\_lfa1-stras '"' INTO  lw\_einv\_api\_hdr-address2\_b.  
                CONCATENATE '"'  lw\_lfa1-ort01 '"'  INTO           lw\_einv\_api\_hdr-location\_b.  
                lw\_einv\_api\_hdr-pincode\_b  = lw\_lfa1-pstlz.  
  
                lw\_einv\_api\_hdr-place\_of\_supply\_b = lw\_lfa1-regio.  
  
                CONCATENATE '"' lw\_einv\_api\_hdr-place\_of\_supply\_b '"' INTO lw\_einv\_api\_hdr-place\_of\_supply\_b.  
                CLEAR lw\_t005u.  
                READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_lfa1-regio.  
\*                  IF sy-subrc EQ 0.  
                TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                CONCATENATE '"' lw\_t005u-bezei '"'  INTO  lw\_einv\_api\_hdr-state\_code\_b.  
                lw\_einv\_api\_hdr-place\_of\_supply\_b   =   lw\_einv\_api\_hdr-state\_code\_b.                                           "State code    Y  
\*                  ENDIF.  
\*                ENDIF.  
              ENDIF.  
              CLEAR lw\_adr61.  
              READ TABLE gt\_adr61 INTO lw\_adr61 WITH KEY addrnumber = lw\_adrc1-addrnumber.  
              CONCATENATE '"' lw\_adr61-smtp\_addr '"'INTO lw\_einv\_api\_hdr-email\_b.  
  
\*\*End of buyer\_details  
\*begin of dispatch\_details  
              lw\_einv\_api\_hdr-comapny\_name\_d = lw\_einv\_api\_hdr-trade\_name\_s.  
              lw\_einv\_api\_hdr-address1\_d = lw\_einv\_api\_hdr-address1\_s.  
              lw\_einv\_api\_hdr-address2\_d  = lw\_einv\_api\_hdr-address2\_s.  
              lw\_einv\_api\_hdr-location\_d = lw\_einv\_api\_hdr-location\_s.  
              lw\_einv\_api\_hdr-pincode\_d = lw\_einv\_api\_hdr-pincode\_s .  
              lw\_einv\_api\_hdr-state\_code\_d = lw\_einv\_api\_hdr-state\_code\_s.  
\*begin of dispatch\_details.  
  
  
\*\*Begin of ship\_details  
  
              lw\_einv\_api\_hdr-gstin\_sh  = lw\_einv\_api\_hdr-gstin\_b.  
              lw\_einv\_api\_hdr-legal\_name\_sh = lw\_einv\_api\_hdr-legal\_name\_b.  
              lw\_einv\_api\_hdr-trade\_name\_sh = lw\_einv\_api\_hdr-trade\_name\_b.  
              lw\_einv\_api\_hdr-address1\_sh = lw\_einv\_api\_hdr-address1\_b.  
              lw\_einv\_api\_hdr-address2\_sh = lw\_einv\_api\_hdr-address2\_b.  
              lw\_einv\_api\_hdr-location\_sh  = lw\_einv\_api\_hdr-location\_b.  
              IF lw\_einv\_api\_hdr-state\_code\_b = 'ANDRA PRADESH'.  
                lw\_einv\_api\_hdr-state\_code\_b = 'ANDHRA PRADESH'.  
              ENDIF.  
              lw\_einv\_api\_hdr-state\_code\_sh  = lw\_einv\_api\_hdr-state\_code\_b.  
              lw\_einv\_api\_hdr-pincode\_sh  = lw\_einv\_api\_hdr-pincode\_b.  
\*End of  ship\_details  
  
\*Begin of export\_details  
  
              CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-ship\_bill\_number.  
              CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-ship\_bill\_date.  
              CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-country\_code.  
              CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-foreign\_currency.  
              CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-refund\_claim.  
              CONCATENATE  '"' ' ' '"' INTO lw\_einv\_api\_hdr-port\_code  .  
\*End of export\_details  
\*\*Begin of payment\_details  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-bank\_account\_number.  
              CONCATENATE ' ' '0' ' ' INTO lw\_einv\_api\_hdr-paid\_balance\_amount.  
              CONCATENATE ' ' '0' ' ' INTO lw\_einv\_api\_hdr-credit\_days.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-credit\_transfer.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-direct\_debit.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-branch\_or\_ifsc.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_mode.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payee\_name.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_due\_date.  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_instruction .  
              CONCATENATE '"' ' ' '"' INTO lw\_einv\_api\_hdr-payment\_term.  
\*\*End of payment\_details  
  
              CONCATENATE  '"' '' '"' INTO lw\_einv\_api\_hdr-invoice\_remarks.  
  
\*Begin of reference\_details  
              lw\_einv\_api\_hdr-invoice\_period\_start\_date = lw\_einv\_api\_hdr-document\_date.  
              lw\_einv\_api\_hdr-invoice\_period\_end\_date   = lw\_einv\_api\_hdr-document\_date.  
  
\*\*Begin of preceding\_document\_details  
              lw\_einv\_api\_hdr-reference\_of\_original\_invoice = lw\_einv\_api\_hdr-document\_number.  
              lw\_einv\_api\_hdr-preceding\_invoice\_date        = lw\_einv\_api\_hdr-document\_date.  
              lw\_einv\_api\_hdr-other\_reference = '""'.  
\*End of preceding\_document\_details  
\*End of reference\_details  
  
\*Begin of contract Details,  
              CONCATENATE '"' '' '"' INTO    lw\_einv\_api\_hdr-receipt\_advice\_number .  
              CONCATENATE '"' '' '"' INTO    lw\_einv\_api\_hdr-batch\_reference\_number.  
              CONCATENATE '"' '' '"' INTO    lw\_einv\_api\_hdr-contract\_reference\_number.  
              CONCATENATE '"' '' '"' INTO    lw\_einv\_api\_hdr-other\_reference\_c.  
              lw\_einv\_api\_hdr-project\_reference\_number = '""'.  
              CONCATENATE '"' '' '"' INTO    lw\_einv\_api\_hdr-vendor\_po\_reference\_number.  
              CONCATENATE '"' '' '"' INTO    lw\_einv\_api\_hdr-vendor\_po\_reference\_date.  
              "End of contract\_details  
  
\*Begin of'"additional\_document\_details"  
              lw\_einv\_api\_hdr-supporting\_document\_url = '""'.  
              lw\_einv\_api\_hdr-supporting\_document     = '""'.  
              lw\_einv\_api\_hdr-additional\_information  = '""'.  
\*End of'"additional\_document\_details"  
  
\*Begin of value\_details  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
              lw\_einv\_api\_hdr-total\_assessable\_value          = '0'.  
              lw\_einv\_api\_hdr-total\_cgst\_value                = '0'.  
              lw\_einv\_api\_hdr-total\_sgst\_value                = '0'.  
              lw\_einv\_api\_hdr-total\_igst\_value                = '0'.  
              lw\_einv\_api\_hdr-total\_cess\_value                = '0'.  
              lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value       = '0'.  
              lw\_einv\_api\_hdr-total\_invoice\_value             = '0'.  
              lw\_einv\_api\_hdr-total\_cess\_value\_of\_state       = '0'.  
              lw\_einv\_api\_hdr-round\_off\_amount                = '0'.  
              lw\_einv\_api\_hdr-total\_invoice\_value\_additional  = '0'.  
              lw\_einv\_api\_hdr-total\_other\_charge              = '0'.  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
              lw\_einv\_api\_hdr-total\_assessable\_value = '1'.               "Required  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*Begin of item\_list  
            ENDIF.  
            LOOP AT gt\_bseg INTO gw\_bseg WHERE bukrs = gw\_bkpf-bukrs  
                                         AND belnr = gw\_bkpf-belnr  
                                         AND gjahr = gw\_bkpf-gjahr  
                                         AND koart = 'S'.  
              "and ktosl ne 'WIT'.  
              IF sy-subrc = 0.  
                "FROM lv\_index.  
\*              IF gw\_bseg-belnr NE gw\_final-vbeln.  
\*                EXIT.  
\*              ENDIF.  
                IF gw\_bseg-buzid IS INITIAL .  
                  "AND gw\_bseg-koart NE 'D'.  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
                  lw\_einv\_api\_itm-quantity                    = '0'.  
                  lw\_einv\_api\_itm-free\_quantity               = '0'.  
                  lw\_einv\_api\_itm-unit\_price                  = '0'.  
                  lw\_einv\_api\_itm-total\_amount                = '0'.  
                  lw\_einv\_api\_itm-pre\_tax\_value               = '0'.  
                  lw\_einv\_api\_itm-discount                    = '0'.  
                  lw\_einv\_api\_itm-other\_charge                = '0'.  
                  lw\_einv\_api\_itm-gst\_rate                    = '0'.  
                  lw\_einv\_api\_itm-igst\_amount                 = '0'.  
                  lw\_einv\_api\_itm-cgst\_amount                 = '0'.  
                  lw\_einv\_api\_itm-sgst\_amount                 = '0'.  
                  lw\_einv\_api\_itm-cess\_rate                   = '0'.  
                  lw\_einv\_api\_itm-cess\_amount                 = '0'.  
                  lw\_einv\_api\_itm-cess\_nonadvol\_value         = '0'.  
                  lw\_einv\_api\_itm-state\_cess\_rate             = '0'.  
                  lw\_einv\_api\_itm-state\_cess\_amount           = '0'.  
                  lw\_einv\_api\_itm-state\_cess\_nonadvol\_amount  = '0'.  
                  lw\_einv\_api\_itm-total\_item\_value            = '0'.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* dafault all values to 0 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
                  lw\_einv\_api\_itm-bukrs  = gw\_bseg-bukrs.  
                  lw\_einv\_api\_itm-doc\_year = gw\_bseg-gjahr.  
                  lw\_einv\_api\_itm-doc\_type = gw\_bkpf-blart.  
                  lw\_einv\_api\_itm-document\_type = gw\_bkpf-blart.  
                  lw\_einv\_api\_itm-document\_number = lw\_einv\_api\_hdr-document\_number   .  
                  SHIFT lw\_einv\_api\_itm-document\_number LEFT DELETING LEADING '0'.  
                  IF gw\_bkpf-budat IS NOT INITIAL.  
                    CONCATENATE  gw\_bkpf-budat+6(2) '/'  
                                 gw\_bkpf-budat+4(2) '/'  
                                 gw\_bkpf-budat+0(4)  
                            INTO lw\_einv\_api\_itm-document\_date.  
                  ENDIF.  
                  CONCATENATE '"' lw\_einv\_api\_itm-document\_type '"' INTO  lw\_einv\_api\_itm-document\_type.  
                  CONCATENATE '"' lw\_einv\_api\_itm-document\_date '"' INTO  lw\_einv\_api\_itm-document\_date.  
  
                  CONCATENATE '"'  gw\_bseg-buzei '"' INTO lw\_einv\_api\_itm-item\_serial\_number .  
                  CONCATENATE '"'  gw\_bseg-sgtxt '"' INTO lw\_einv\_api\_itm-product\_description .  
  
  
                  IF gw\_bseg-matnr IS INITIAL.  
                    lw\_einv\_api\_itm-is\_service = gc\_service\_y.  
                  ELSE.  
                    lw\_einv\_api\_itm-is\_service = gc\_service\_n.  
                  ENDIF.  
  
                  lw\_einv\_api\_itm-hsn\_code = gw\_bseg-hsn\_sac.  
                  "added by lakshmi.  
                  DATA(len) = strlen( lw\_einv\_api\_itm-hsn\_code ).  
                  IF len = '4' OR len = '6'.  
                    IF lw\_einv\_api\_itm-hsn\_code(2) = '99'.  
                      lw\_einv\_api\_itm-is\_service = gc\_service\_y.  
                    ELSE.  
                      lw\_einv\_api\_itm-is\_service = gc\_service\_n.  
                    ENDIF.  
                  ELSE.  
                    lw\_einv\_api\_itm-is\_service = gc\_service\_n.  
                  ENDIF.  
                  "added by lakshmi.  
                  IF lw\_einv\_api\_itm-hsn\_code IS INITIAL.  
\*                  lw\_einv\_api\_itm-hsn\_code  = ''.  
                  ENDIF.  
                  CONCATENATE '"' lw\_einv\_api\_itm-hsn\_code '"' INTO lw\_einv\_api\_itm-hsn\_code.  
                  lw\_einv\_api\_itm-bar\_code =  '""'.  
  
                  CONCATENATE '"'  '"' INTO lw\_einv\_api\_itm-order\_line\_reference.  
                  CONCATENATE '"'  '"' INTO lw\_einv\_api\_itm-product\_serial\_number.  
  
\*Begin of  attribute\_details"  
                  lw\_einv\_api\_itm-item\_attribute\_details = gc\_attr\_einv.  
                  lw\_einv\_api\_itm-item\_attribute\_value = gc\_attr\_einv.  
\*End of of  attribute\_details"  
  
\*Begin of  attribute\_details"  
                  lw\_einv\_api\_itm-item\_attribute\_details = gc\_attr\_einv.  
                  lw\_einv\_api\_itm-item\_attribute\_value = gc\_attr\_einv.  
\*End of of  attribute\_details"  
\*End of item\_list  
  
                  "Quantity  
                  lw\_einv\_api\_itm-quantity = gw\_bseg-menge.  
                  IF lw\_einv\_api\_itm-quantity IS INITIAL.  
                    lw\_einv\_api\_itm-quantity = '1'.  
                  ENDIF.  
                  lw\_einv\_api\_itm-free\_quantity = '0'.  
  
\*Unit  
                  CLEAR lv\_output\_uom.  
                  IF gw\_bseg-meins IS NOT INITIAL.  
                    CALL FUNCTION 'CONVERSION\_EXIT\_CUNIT\_OUTPUT'  
                      EXPORTING  
                        input          = gw\_bseg-meins  
                      IMPORTING  
                        output         = lv\_output\_uom  
                      EXCEPTIONS  
                        unit\_not\_found = 1  
                        OTHERS         = 2.  
                    IF sy-subrc <> 0.  
\* Implement suitable error handling here  
                    ENDIF.  
  
                    READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins = lv\_output\_uom.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_einv\_api\_itm-unit.  
                    ELSE.  
                      READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins =  gw\_bseg-meins.  
                      IF sy-subrc IS INITIAL.  
                        CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_einv\_api\_itm-unit.  
                      ENDIF.  
                    ENDIF.  
                  ELSE.  
                    lw\_einv\_api\_itm-unit = '"OTH"'.  
                  ENDIF.  
                  lw\_einv\_api\_itm-other\_charge =  '0'.  
                  IF gw\_bseg-buzid IS INITIAL.  
                    IF gw\_bseg-menge IS NOT INITIAL.  
                      lv\_rate = gw\_bseg-dmbtr / gw\_bseg-menge.  
                      lw\_einv\_api\_itm-unit\_price = lv\_rate.  
                    ELSE.  
                      lw\_einv\_api\_itm-unit\_price = '1.00'.  
                    ENDIF.  
                    lw\_einv\_api\_itm-total\_amount = gw\_bseg-dmbtr.  
                    lw\_einv\_api\_itm-unit\_price = lw\_einv\_api\_itm-total\_amount."added by lakshmi  
                    lw\_einv\_api\_itm-assessable\_value = gw\_bseg-dmbtr.  
                  ENDIF.  
                  CONDENSE:lw\_einv\_api\_itm-unit\_price,lw\_einv\_api\_itm-total\_amount,  
                            lw\_einv\_api\_itm-assessable\_value.  
                  lv\_total\_assessable\_value = lv\_total\_assessable\_value  + lw\_einv\_api\_itm-assessable\_value.  
                  READ TABLE gt\_bseg INTO lw\_bseg WITH KEY bukrs = gw\_bseg-bukrs  
                                                           belnr = gw\_bseg-belnr  
                                                           gjahr = gw\_bseg-gjahr  
                                                           txgrp = gw\_bseg-txgrp  
                                                           ktosl = gc\_ktosl\_tcs.  
                  IF sy-subrc = 0 .  
                    lw\_einv\_api\_itm-other\_charge = lw\_einv\_api\_itm-other\_charge + lw\_bseg-dmbtr.  
                  ENDIF.  
                  CLEAR:lw\_bseg.  
                  READ TABLE gt\_bseg INTO lw\_bseg WITH KEY bukrs = gw\_bseg-bukrs  
                                                           belnr = gw\_bseg-belnr  
                                                           gjahr = gw\_bseg-gjahr  
                                                           txgrp = gw\_bseg-txgrp  
                                                           ktosl = gc\_ktosl\_jos.  
                  IF sy-subrc = 0 .  
                    lw\_einv\_api\_itm-sgst\_amount = lw\_bseg-dmbtr.  
                    IF lw\_bseg-dmbtr IS NOT INITIAL.  
                      lw\_einv\_api\_itm-gst\_rate = ( lw\_bseg-dmbtr / gw\_bseg-dmbtr ) \* 100.  
                    ENDIF.  
                    CONDENSE:lw\_einv\_api\_itm-gst\_rate,lw\_einv\_api\_itm-sgst\_amount.  
                    ADD lw\_einv\_api\_itm-gst\_rate TO lw\_gstrate.  
                    ADD lw\_einv\_api\_itm-sgst\_amount TO lv\_total\_sgst\_value.  
                  ENDIF.  
                  READ TABLE gt\_bseg INTO lw\_bseg WITH KEY bukrs = gw\_bseg-bukrs  
                                                  belnr = gw\_bseg-belnr  
                                                  gjahr = gw\_bseg-gjahr  
                                                  txgrp = gw\_bseg-txgrp  
                                                  ktosl = gc\_ktosl\_joc.  
                  IF sy-subrc = 0.  
  
                    lw\_einv\_api\_itm-cgst\_amount = lw\_bseg-dmbtr.  
                    IF lw\_bseg-dmbtr IS NOT INITIAL.  
                      lw\_einv\_api\_itm-gst\_rate = ( lw\_bseg-dmbtr / gw\_bseg-dmbtr ) \* 100.  
                    ENDIF.  
                    CONDENSE:lw\_einv\_api\_itm-gst\_rate,lw\_einv\_api\_itm-cgst\_amount.  
                    ADD lw\_einv\_api\_itm-gst\_rate TO lw\_gstrate.  
                    ADD lw\_einv\_api\_itm-cgst\_amount TO lv\_total\_cgst\_value.  
                  ENDIF.  
                  READ TABLE gt\_bseg INTO lw\_bseg WITH KEY bukrs = gw\_bseg-bukrs  
                                            belnr = gw\_bseg-belnr  
                                            gjahr = gw\_bseg-gjahr  
                                            ktosl = gc\_ktosl\_joi.  
                  IF sy-subrc IS INITIAL.  
                    lw\_einv\_api\_itm-igst\_amount = lw\_bseg-dmbtr.  
                    IF lw\_bseg-dmbtr IS NOT INITIAL.  
                      lw\_einv\_api\_itm-gst\_rate = ( lw\_bseg-dmbtr / gw\_bseg-dmbtr ) \* 100.  
                    ENDIF.  
                    CONDENSE:lw\_einv\_api\_itm-gst\_rate,lw\_einv\_api\_itm-igst\_amount.  
                    ADD lw\_einv\_api\_itm-gst\_rate TO lw\_gstrate.  
                    ADD lw\_einv\_api\_itm-igst\_amount TO lv\_total\_igst\_value.  
                  ENDIF.  
  
                  lw\_einv\_api\_itm-total\_item\_value =  ( lw\_einv\_api\_itm-assessable\_value +  
                                                        lw\_einv\_api\_itm-cgst\_amount +  
                                                        lw\_einv\_api\_itm-sgst\_amount +  
                                                        "lw\_einv\_api\_itm-cess\_amount +  
                                                        lw\_einv\_api\_itm-igst\_amount +  
                                                        lw\_einv\_api\_itm-state\_cess\_amount +  
                                                        lw\_einv\_api\_itm-cess\_nonadvol\_value +  
                                                        lw\_einv\_api\_itm-other\_charge ).  
                  CONDENSE:lw\_einv\_api\_itm-total\_item\_value.  
                  ADD lw\_einv\_api\_itm-total\_item\_value TO  lv\_total\_invoice\_value .  
  
                  lw\_einv\_api\_itm-gst\_rate = lw\_gstrate.  
                  CONDENSE:lw\_einv\_api\_itm-gst\_rate.  
                  APPEND lw\_einv\_api\_itm TO gt\_einv\_api\_itm.  
                  CLEAR:lw\_einv\_api\_itm,lw\_gstrate.  
                ENDIF.  
              ENDIF.  
            ENDLOOP.  
\*End of item\_list  
            lw\_einv\_api\_hdr-total\_assessable\_value = lv\_total\_assessable\_value.  
            lw\_einv\_api\_hdr-total\_cgst\_value       = lv\_total\_cgst\_value.  
            lw\_einv\_api\_hdr-total\_sgst\_value       = lv\_total\_sgst\_value.  
            lw\_einv\_api\_hdr-total\_igst\_value       = lv\_total\_igst\_value .  
            lw\_einv\_api\_hdr-total\_invoice\_value    = lv\_total\_invoice\_value.  
            lw\_einv\_api\_hdr-total\_cess\_value       = lv\_total\_cess\_value.  
            lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value =  lv\_total\_cess\_nonadvol\_value.  
            CONDENSE:lw\_einv\_api\_hdr-total\_assessable\_value,lw\_einv\_api\_hdr-total\_cgst\_value,lw\_einv\_api\_hdr-total\_sgst\_value,lw\_einv\_api\_hdr-total\_igst\_value,  
                     lw\_einv\_api\_hdr-total\_invoice\_value,lw\_einv\_api\_hdr-total\_cess\_value,lw\_einv\_api\_hdr-total\_cess\_nonadvol\_value.  
            APPEND lw\_einv\_api\_hdr TO gt\_einv\_api\_hdr.  
            CLEAR:lw\_einv\_api\_hdr.  
  
  
          ENDIF.  
\*          ENDIF.  
        ELSE.  
          MESSAGE 'Please select appropriate documents for E-Doc generation'(039) TYPE 'I'.  
  
        ENDIF.  
      ENDLOOP.  
  
      IF gt\_einv\_api\_hdr IS NOT INITIAL.  
        SORT gt\_einv\_api\_hdr .  
        SORT gt\_einv\_api\_itm .  
  
        DELETE ADJACENT DUPLICATES FROM   gt\_einv\_api\_hdr  COMPARING ALL FIELDS.  
        DELETE ADJACENT DUPLICATES FROM  gt\_einv\_api\_itm  COMPARING ALL FIELDS.  
  
  
  
        lt\_api\_hdr = gt\_einv\_api\_hdr .  
        lt\_api\_itm = gt\_einv\_api\_itm .  
  
  
  
        CLEAR:lw\_token,lw\_return.  
        IF gw\_token IS NOT INITIAL.  
          lw\_token = gw\_token.  
        ELSE.  
          CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
            IMPORTING  
              ex\_token    = lw\_token  
              ex\_return   = lw\_return  
              et\_messages = lt\_messages.  
        ENDIF.  
        IF lw\_token IS NOT INITIAL." AND lw\_return EQ 'S'.  
          CALL FUNCTION 'ZFM\_EINVOICE\_GENERATE\_API'  
            EXPORTING  
              im\_token        = lw\_token  
              im\_api\_hdr      = lt\_api\_hdr  
              im\_api\_itm      = lt\_api\_itm  
            IMPORTING  
              ex\_return       = lw\_return  
              ex\_invref       = lt\_invref  
              ex\_messages     = lt\_messages  
              ex\_einv\_details = lt\_einv\_details.  
  
          IF lt\_invref IS NOT INITIAL."lw\_return EQ 'S'.  
            MODIFY j\_1ig\_invrefnum FROM TABLE lt\_invref.  
            IF sy-subrc EQ 0.  
              MODIFY zteinv\_details FROM TABLE lt\_einv\_details.  
  
              LOOP AT lt\_invref INTO lw\_invref.  
                READ TABLE gt\_final ASSIGNING <gw\_final> WITH KEY vbeln = lw\_invref-docno.  
                IF sy-subrc IS INITIAL.  
  
                  <gw\_final>-irn = lw\_invref-irn.  
                  <gw\_final>-ack\_no = lw\_invref-ack\_no.  
                  IF lw\_invref-ack\_date IS NOT INITIAL.  
                    lv\_date = lw\_invref-ack\_date.  
                    CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                      EXPORTING  
                        date\_internal            = lv\_date  
                      IMPORTING  
                        date\_external            = <gw\_final>-ack\_dt  
                      EXCEPTIONS  
                        date\_internal\_is\_invalid = 1  
                        OTHERS                   = 2.  
                    IF sy-subrc <> 0.  
                      MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
                    ENDIF.  
                  ENDIF.  
  
                  <gw\_final>-ernam = lw\_invref-ernam.  
  
                  CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                    EXPORTING  
                      date\_internal            = lw\_invref-erdat  
                    IMPORTING  
                      date\_external            = <gw\_final>-erdat  
                    EXCEPTIONS  
                      date\_internal\_is\_invalid = 1  
                      OTHERS                   = 2.  
                  IF sy-subrc <> 0.  
                    MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
                  ENDIF.  
  
                  <gw\_final>-erzet = |{ lw\_invref-erzet(2) }:{ lw\_invref-erzet+2(2) }:{ lw\_invref-erzet+4(2) }|.  
  
  
                  IF lw\_invref-irn\_status = gc\_irn\_sts\_act.  
                    <gw\_final>-status = 'Success'(060).  
                    <gw\_final>-icon    = gc\_icon\_08.  
                    <gw\_final>-einv\_error = space.  
                  ELSEIF lw\_invref-irn\_status = gc\_irn\_sts\_err.  
                    <gw\_final>-status = 'Error'(061).  
                    <gw\_final>-icon    = gc\_icon\_0a.  
                    READ TABLE lt\_einv\_details INTO lw\_einv\_details WITH KEY docno = lw\_invref-docno.  
                    IF  sy-subrc IS INITIAL.  
                      <gw\_final>-einv\_error = lw\_einv\_details-einv\_error.  
                    ENDIF.  
                  ELSEIF lw\_invref-irn\_status = gc\_irn\_sts\_cnl.  
                    <gw\_final>-status = 'Cancelled'(062).  
                    <gw\_final>-icon    =  gc\_icon\_0w.  
                  ENDIF.  
                ENDIF.  
  
              ENDLOOP.  
  
  
  
\* COmmit work  
              COMMIT WORK.  
  
  
              CALL METHOD gref\_alv\_grid->refresh\_table\_display  
                EXCEPTIONS  
                  finished = 1  
                  OTHERS   = 2.  
              IF sy-subrc <> 0.  
                MESSAGE 'Table refresh error'(012) TYPE 'I'.  
              ENDIF.  
            ENDIF.  
          ENDIF.  
        ENDIF.  
  
  
        CLEAR lt\_show\_message.  
        IF  lt\_messages IS NOT INITIAL.  
  
          LOOP AT lt\_messages INTO lw\_message.  
            IF lw\_message-type IS NOT INITIAL.  
              lw\_show\_message-msgid =  gc\_msgid\_01.  
              lw\_show\_message-msgty =  lw\_message-type.  
              lw\_show\_message-msgno =  gc\_msgno\_319.  
              lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
              lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
              lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
              lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
              APPEND lw\_show\_message TO lt\_show\_message.  
            ENDIF.  
          ENDLOOP.  
  
          CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
            TABLES  
              i\_message\_tab = lt\_show\_message.  
        ENDIF.  
      ELSE.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
ENDFORM.  
\*---------------------------------------------------------------------\*  
\*       FORM EXIT\_PROGRAM                                             \*  
\*---------------------------------------------------------------------\*  
FORM exit\_program.  
  CALL METHOD gref\_alv\_container->free.  
  LEAVE TO SCREEN 0.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EINVOICE\_CANCEL  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM einvoice\_cancel USING p\_lv\_reason\_code  
                           p\_lv\_reason.  
  
  DATA:lw\_token         TYPE string,  
       lw\_return        TYPE string,  
       wa\_return        TYPE string,  
       lt\_messages      TYPE bapiret2\_t,  
       lw\_message       TYPE bapiret2,  
       lt\_show\_message  TYPE esp1\_message\_tab\_type,  
       lw\_show\_message  LIKE LINE OF lt\_show\_message,  
       lt\_invref        TYPE TABLE OF j\_1ig\_invrefnum,  
       lw\_invref        TYPE  j\_1ig\_invrefnum,  
       lw\_index         LIKE LINE OF gt\_index,  
       lv\_val           TYPE xfeld,  
       lv\_answer        TYPE c,  
       lv\_gstin         TYPE  kna1-stcd3,  
       lv\_reason        TYPE  char20,  
       lv\_remarks       TYPE  char255,  
       ls\_values        TYPE dd07v,  
       lv\_internal\_date TYPE sy-datum,  
       lw\_usrgstin      TYPE ty\_api.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  
  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
  
    LOOP AT  gt\_index INTO lw\_index.  
  
      READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
      IF sy-subrc IS INITIAL AND gw\_final-irn IS NOT INITIAL.  
        lv\_gstin = gw\_final-sup\_gstin.  
        SELECT \* FROM j\_1ig\_invrefnum APPENDING TABLE lt\_invref  
                                      WHERE bukrs     = gw\_final-bukrs AND  
                                            docno     = gw\_final-vbeln AND  
                                            doc\_type  = gw\_final-fkart AND  
                                            irn       = gw\_final-irn.  
  
      ENDIF.  
    ENDLOOP.  
  
    READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_usrgstin.  
    IF sy-subrc IS INITIAL.  
      lv\_gstin   = lw\_usrgstin-apiprov.  
    ENDIF.  
  
  
  
    IF lt\_invref IS NOT INITIAL.  
      CLEAR:lw\_token,lw\_return.  
      IF gw\_token IS NOT INITIAL.  
        lw\_token = gw\_token.  
      ELSE.  
  
        CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
          IMPORTING  
            ex\_token    = lw\_token  
            ex\_return   = lw\_return  
            et\_messages = lt\_messages.  
      ENDIF.  
      IF lw\_token IS NOT INITIAL.  
        CONCATENATE '"' p\_lv\_reason\_code '"' INTO lv\_reason.  
        CONCATENATE '"' p\_lv\_reason       '"' INTO lv\_remarks.  
        CONCATENATE '"' lv\_gstin '"' INTO lv\_gstin.  
  
  
        CALL FUNCTION 'ZFM\_EINVOICE\_CANCEL\_API'  
          EXPORTING  
            im\_token    = lw\_token  
            im\_invref   = lt\_invref  
            im\_gstin    = lv\_gstin  
            im\_reason   = lv\_reason  
            im\_remarks  = lv\_remarks  
          IMPORTING  
            ex\_return   = lw\_return  
            ex\_invref   = lt\_invref  
            ex\_messages = lt\_messages.  
  
        IF lt\_invref IS NOT INITIAL."lw\_return EQ 'S'.  
          MODIFY j\_1ig\_invrefnum FROM TABLE lt\_invref.  
  
          LOOP AT lt\_invref INTO lw\_invref.  
            READ TABLE gt\_final ASSIGNING <gw\_final> WITH KEY vbeln = lw\_invref-docno.  
            IF sy-subrc IS INITIAL.  
  
              IF lw\_invref-cancel\_date IS NOT INITIAL.  
                lw\_invref-irn\_status     = gc\_irn\_sts\_cnl.  
  
                UPDATE zteinv\_details SET e\_reason\_code = p\_lv\_reason\_code  
                                            e\_reason      = p\_lv\_reason  
                                            aenam         = sy-uname  
                                            aedat         = sy-datum  
                                          WHERE  bukrs  = lw\_invref-bukrs AND  
                                            docno  = lw\_invref-docno AND  
                                            doctyp = lw\_invref-doc\_type AND  
                                            gjahr  = lw\_invref-doc\_year.  
  
  
                IF sy-subrc IS INITIAL.  
                  lv\_internal\_date = lw\_invref-cancel\_date.  
                  CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                    EXPORTING  
                      date\_internal            = lv\_internal\_date  
                    IMPORTING  
                      date\_external            = <gw\_final>-canc\_dt  
                    EXCEPTIONS  
                      date\_internal\_is\_invalid = 1  
                      OTHERS                   = 2.  
                  IF sy-subrc <> 0.  
                    MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
                  ENDIF.  
                ENDIF.  
\*                  ENDIF.  
              ENDIF.  
  
              IF lw\_invref-irn\_status = gc\_irn\_sts\_act.  
                <gw\_final>-status = 'Success'(060).  
                <gw\_final>-icon    = gc\_icon\_08.  
              ELSEIF lw\_invref-irn\_status = gc\_irn\_sts\_err.  
                <gw\_final>-status = 'Error'(061).  
                <gw\_final>-icon    = gc\_icon\_0a.  
              ELSEIF lw\_invref-irn\_status = gc\_irn\_sts\_cnl.  
                <gw\_final>-status = 'Cancelled'(062).  
                <gw\_final>-icon    =  gc\_icon\_0w.  
              ENDIF.  
            ENDIF.  
          ENDLOOP.  
  
\* Commit work  
          COMMIT WORK.  
  
          CALL METHOD gref\_alv\_grid->refresh\_table\_display  
            EXCEPTIONS  
              finished = 1  
              OTHERS   = 2.  
          IF sy-subrc <> 0.  
            MESSAGE 'Table refresh error'(012) TYPE 'I'.  
          ENDIF.  
  
  
\*            ENDIF.  
        ENDIF.  
  
      ENDIF.  
  
  
      CLEAR lt\_show\_message.  
      IF  lt\_messages IS NOT INITIAL.  
        LOOP AT lt\_messages INTO lw\_message.  
          lw\_show\_message-msgid =  gc\_msgid\_01.  
          lw\_show\_message-msgty =  lw\_message-type.  
          lw\_show\_message-msgno =  gc\_msgno\_319.  
          lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
          lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
          lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
          lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
          APPEND lw\_show\_message TO lt\_show\_message.  
  
        ENDLOOP.  
  
        CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
          TABLES  
            i\_message\_tab = lt\_show\_message.  
  
      ENDIF.  
    ELSE.  
      MESSAGE 'Please select appropriate documents for E-Invoice cancellation'(014) TYPE 'I'.  
    ENDIF.  
  
  
  
  ENDIF.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  CALL\_POPUP  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->P\_TEXT\_T20  text  
\*      -->P\_TEXT\_T21  text  
\*----------------------------------------------------------------------\*  
FORM call\_popup  USING    p\_text\_t20  
                          p\_text\_t21  
                   CHANGING p\_text\_t22 TYPE c.  
  
  
  CALL FUNCTION 'POPUP\_TO\_CONFIRM'  
    EXPORTING  
      titlebar              = p\_text\_t20  
      text\_question         = p\_text\_t21  
      text\_button\_1         = 'Yes'(107)  
      text\_button\_2         = 'No'(108)  
      display\_cancel\_button = abap\_true  
    IMPORTING  
      answer                = p\_text\_t22  
    EXCEPTIONS  
      text\_not\_found        = 1  
      OTHERS                = 2.  
  IF sy-subrc <> 0.  
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
            WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
  ENDIF.  
  
  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Module  USER\_COMMAND\_9001  INPUT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
MODULE user\_command\_9001 INPUT.  
  
  DATA:lw\_index  LIKE LINE OF gt\_index.  
  DATA:lv\_answer TYPE c.  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  DATA: lt\_zteway\_transport TYPE TABLE OF zteway\_transport,  
        ls\_zteway\_transport TYPE zteway\_transport.  
  
  CASE sy-ucomm.  
  
    WHEN 'OK'.  
  
      PERFORM call\_popup USING   TEXT-t23 TEXT-t30  
                         CHANGING lv\_answer.  
  
      IF lv\_answer = '1'.  
  
        LOOP AT  gt\_index INTO lw\_index.  
          READ TABLE gt\_final ASSIGNING <gw\_final> INDEX lw\_index-index.  
          IF sy-subrc IS INITIAL.  
  
            <gw\_final>-t\_id      = zst\_einv\_api\_struct-transporter\_id.  
            <gw\_final>-t\_name    = zst\_einv\_api\_struct-transporter\_name.  
            <gw\_final>-t\_mode = zst\_einv\_api\_struct-transportation\_mode.  
            <gw\_final>-t\_distance   = zst\_einv\_api\_struct-transportation\_distance.  
            <gw\_final>-t\_doc\_no  = zst\_einv\_api\_struct-transporter\_document\_number.  
            IF zst\_einv\_api\_struct-transporter\_document\_date IS NOT INITIAL.  
              CONCATENATE  zst\_einv\_api\_struct-transporter\_document\_date+6(2) '/'  
                           zst\_einv\_api\_struct-transporter\_document\_date+4(2) '/'  
                           zst\_einv\_api\_struct-transporter\_document\_date+0(4)  
                           INTO <gw\_final>-t\_date.  
            ENDIF.  
            CONDENSE zst\_einv\_api\_struct-vehicle\_number NO-GAPS.  
            <gw\_final>-v\_number  = zst\_einv\_api\_struct-vehicle\_number.  
            <gw\_final>-v\_type = zst\_einv\_api\_struct-vehicle\_type.  
            <gw\_final>-t\_r\_distance           = zst\_einv\_api\_struct-remaining\_distance.  
            <gw\_final>-t\_ext\_valid\_reason = zst\_einv\_api\_struct-vehicle\_type.  
            <gw\_final>-t\_ext\_valid\_remarks    = zst\_einv\_api\_struct-extend\_remarks.  
            <gw\_final>-t\_from\_pin             = zst\_einv\_api\_struct-from\_pincode.  
            <gw\_final>-t\_consignment\_status   = zst\_einv\_api\_struct-consignment\_status.  
            <gw\_final>-t\_transit\_type         = zst\_einv\_api\_struct-transit\_type.  
            IF zst\_einv\_api\_struct-transportation\_mode NE gc\_tmode\_5.  
              <gw\_final>-t\_transit\_type         = space.  
            ENDIF.  
            <gw\_final>-t\_address1             = zst\_einv\_api\_struct-address\_line1.  
            <gw\_final>-t\_address2             = zst\_einv\_api\_struct-address\_line2.  
            <gw\_final>-t\_address3             = zst\_einv\_api\_struct-address\_line3.  
            <gw\_final>-v\_reason\_code = zst\_einv\_api\_struct-reason\_code\_for\_vehicle\_updati.  
            <gw\_final>-v\_reason  = zst\_einv\_api\_struct-reason\_for\_vehicle\_updation.  
            <gw\_final>-zport\_code  = zst\_einv\_api\_struct-zport\_code.  
          ENDIF.  
  
          ls\_zteway\_transport-bukrs        = <gw\_final>-bukrs.  
          ls\_zteway\_transport-doctyp       = <gw\_final>-fkart.  
          ls\_zteway\_transport-docno        = <gw\_final>-vbeln.  
          ls\_zteway\_transport-gjahr        = <gw\_final>-gjahr.  
          ls\_zteway\_transport-t\_id         = <gw\_final>-t\_id.  
          ls\_zteway\_transport-t\_name       = <gw\_final>-t\_name.  
          ls\_zteway\_transport-t\_doc\_no     = <gw\_final>-t\_doc\_no.  
          ls\_zteway\_transport-t\_date       = zst\_einv\_api\_struct-transporter\_document\_date.  
          ls\_zteway\_transport-t\_mode       = <gw\_final>-t\_mode.  
          ls\_zteway\_transport-t\_distance   = <gw\_final>-t\_distance.  
          ls\_zteway\_transport-v\_number     = <gw\_final>-v\_number.  
          ls\_zteway\_transport-v\_type       = <gw\_final>-v\_type.  
          ls\_zteway\_transport-t\_r\_distance          = <gw\_final>-t\_r\_distance.  
          ls\_zteway\_transport-t\_ext\_valid\_reason    = <gw\_final>-t\_ext\_valid\_reason.  
          ls\_zteway\_transport-t\_ext\_valid\_remarks   = <gw\_final>-t\_ext\_valid\_remarks.  
          ls\_zteway\_transport-t\_from\_pin            = <gw\_final>-t\_from\_pin.  
          ls\_zteway\_transport-t\_consignment\_status  = <gw\_final>-t\_consignment\_status.  
          ls\_zteway\_transport-t\_transit\_type        = <gw\_final>-t\_transit\_type.  
          ls\_zteway\_transport-t\_address1            = <gw\_final>-t\_address1.  
          ls\_zteway\_transport-t\_address2            = <gw\_final>-t\_address2.  
          ls\_zteway\_transport-t\_address3            = <gw\_final>-t\_address3.  
          ls\_zteway\_transport-v\_reason\_code        = <gw\_final>-v\_reason\_code.  
          ls\_zteway\_transport-v\_reason             = <gw\_final>-v\_reason.  
          ls\_zteway\_transport-zport\_code           = <gw\_final>-zport\_code.  
          ls\_zteway\_transport-eway\_print           = <gw\_final>-eway\_print.  
          ls\_zteway\_transport-eway\_error           = <gw\_final>-eway\_error.  
  
          READ TABLE gt\_zteway\_transport TRANSPORTING NO FIELDS WITH KEY bukrs = <gw\_final>-bukrs  
                                                                         doctyp = <gw\_final>-fkart  
                                                                         docno = <gw\_final>-vbeln.  
          IF sy-subrc IS INITIAL.  
            ls\_zteway\_transport-aenam        = sy-uname.  
            ls\_zteway\_transport-aedat        = sy-datum.  
  
          ELSE.  
            ls\_zteway\_transport-ernam        = sy-uname.  
            ls\_zteway\_transport-erdat        = sy-datum.  
          ENDIF.  
  
  
\*          APPEND ls\_zteway\_transport TO lt\_zteway\_transport.  
          MODIFY zteway\_transport FROM ls\_zteway\_transport.  
          IF sy-subrc IS NOT INITIAL.  
            MESSAGE 'Data update error'(022) TYPE 'I'.  
          ENDIF.  
        ENDLOOP.  
  
        CALL METHOD gref\_alv\_grid->refresh\_table\_display  
          EXCEPTIONS  
            finished = 1  
            OTHERS   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Table refresh error'(012) TYPE 'I'.  
        ENDIF.  
        PERFORM set\_selected\_data.  
  
        LEAVE TO SCREEN 0.  
      ENDIF.  
    WHEN 'CANCEL'.  
      LEAVE TO SCREEN 0.  
  ENDCASE.  
  
ENDMODULE.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_GENERATE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_generate.  
  
  DATA:lw\_index            TYPE lvc\_s\_row,  
       lv\_val              TYPE xfeld,  
       lv\_answer           TYPE c,  
       lv\_index            TYPE sy-tabix,  
       lw\_token            TYPE string,  
       lw\_return           TYPE string,  
       wa\_return           TYPE string,  
       lt\_messages         TYPE bapiret2\_t,  
       lw\_message          TYPE bapiret2,  
       lt\_show\_message     TYPE esp1\_message\_tab\_type,  
       lw\_show\_message     LIKE LINE OF lt\_show\_message,  
       lw\_etransport       TYPE zteway\_transport,  
       lt\_ewaybill         TYPE TABLE OF j\_1ig\_ewaybill,  
       lw\_ewaybill         TYPE j\_1ig\_ewaybill,  
       lt\_eway\_transport   TYPE TABLE OF zteway\_transport,  
       ls\_eway\_transport   TYPE zteway\_transport,  
       lw\_wb2\_v\_vbrk\_vbrp2 TYPE ty\_wb2\_v\_vbrk\_vbrp2,  
       lw\_wb2\_v\_likp\_lips2 TYPE ty\_wb2\_v\_likp\_lips2,  
       ls\_doc\_uom          TYPE ty\_doc\_uom,  
       ls\_doctyp           TYPE ty\_doctyp,  
       lv\_output\_uom       TYPE meins,  
       lt\_api\_hdr          TYPE ztt\_eway\_api\_struct,  
       lt\_api\_itm          TYPE ztt\_eway\_api\_struct\_itm,  
       lw\_eway\_api\_hdr     TYPE zst\_eway\_api\_struct,  
       lw\_eway\_api\_itm     TYPE zst\_eway\_api\_struct\_itm,  
       lw\_t001w            TYPE ty\_t001w1,  
       lw\_t005u\_s          TYPE ty\_t005u,  
       lw\_kna1             TYPE ty\_kna1,  
       lw\_kna1\_sh          TYPE ty\_kna1,  
       lw\_t005u            TYPE ty\_t005u,  
       lw\_adrc             TYPE ty\_adrc,  
       lw\_adr6             TYPE ty\_adr6,  
       lw\_marc             TYPE ty\_marc,  
       ls\_export           TYPE ty\_export,  
       lw\_vbpa             TYPE ty\_vbpa,  
       lw\_usrgstin         TYPE ty\_api,  
       lw\_konv             TYPE ty\_konv,  
       ls\_values           TYPE dd07v,  
       lv\_error            TYPE xfeld,  
       lv\_string           TYPE string,  
       lv\_vbeln            TYPE vbrk-vbeln,  
       lv\_export           TYPE xfeld,  
       lv\_exp\_curr         TYPE xfeld.  
  
  DATA: lv\_supply\_type     TYPE char10 VALUE 'O',  
        lv\_document\_type   TYPE char10 VALUE 1,  
        lv\_sub\_supply\_type TYPE char10 VALUE 1,  
        lv\_document\_number TYPE i VALUE 2.  
  
  DATA: lv\_total\_invoice\_value   TYPE netwr,  
        lv\_total\_sgst\_value      TYPE netwr,  
        lv\_total\_cgst\_value      TYPE netwr,  
        lv\_total\_igst\_value      TYPE netwr,  
        lv\_total\_cess\_value      TYPE netwr,  
        lv\_total\_cessnoval\_value TYPE netwr,  
        lv\_total\_taxable\_value   TYPE netwr.  
  
  FIELD-SYMBOLS: <gw\_final>    TYPE ty\_final,  
                 <lw\_ewaybill> TYPE j\_1ig\_ewaybill.  
  
  CLEAR: lt\_messages,lt\_ewaybill,lt\_api\_hdr,lt\_api\_itm,gt\_eway\_api\_hdr,gt\_eway\_api\_itm,lt\_show\_message.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
    CLEAR lv\_error.  
  
    IF lv\_error IS INITIAL.  
      PERFORM call\_popup USING   TEXT-t23 TEXT-t24  
                         CHANGING lv\_answer.  
  
      IF lv\_answer = '1'.  
        LOOP AT  gt\_index INTO lw\_index.  
  
          READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
          IF sy-subrc IS INITIAL AND ( gw\_final-eway\_num IS INITIAL OR gw\_final-eway\_status = 'Cancelled'(062) )  
             AND gw\_final-eway = abap\_true.  
            IF p\_mod = gc\_sd.  
              READ TABLE gt\_wb2\_v\_vbrk\_vbrp2 INTO lw\_wb2\_v\_vbrk\_vbrp2 WITH KEY vbeln = gw\_final-vbeln.  
              IF sy-subrc = 0.  
                IF lw\_wb2\_v\_vbrk\_vbrp2-fksto NE abap\_true.  
                  lv\_index = sy-tabix.  
  
\*\*\* Reset header value for every document  
                  lv\_total\_invoice\_value   = 0.  
                  lv\_total\_sgst\_value      = 0.  
                  lv\_total\_cgst\_value      = 0.  
                  lv\_total\_igst\_value      = 0.  
                  lv\_total\_cess\_value      = 0.  
                  lv\_total\_cessnoval\_value = 0.  
                  lv\_total\_taxable\_value    = 0.  
  
  
                  LOOP AT gt\_wb2\_v\_vbrk\_vbrp2 INTO lw\_wb2\_v\_vbrk\_vbrp2 FROM lv\_index.  
                    IF lw\_wb2\_v\_vbrk\_vbrp2-vbeln NE gw\_final-vbeln.  
                      EXIT.  
                    ENDIF.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
                    lw\_eway\_api\_hdr-bukrs       = lw\_wb2\_v\_vbrk\_vbrp2-bukrs.  
  
                    READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_eusrgstin.  
                    IF sy-subrc IS INITIAL.  
                      CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO lw\_eway\_api\_hdr-user\_gstin.  
                    ELSE.  
                      CONCATENATE '"' gw\_final-sup\_gstin '"' INTO lw\_eway\_api\_hdr-user\_gstin.  
                    ENDIF.  
  
                    READ TABLE gt\_doctyp INTO ls\_doctyp WITH KEY fkart = lw\_wb2\_v\_vbrk\_vbrp2-fkart.  
                    IF sy-subrc EQ 0.  
                      lv\_supply\_type = ls\_doctyp-sup\_type.  
                      lv\_sub\_supply\_type = ls\_doctyp-sub\_type.  
                      lv\_document\_type = ls\_doctyp-edoc\_type.  
                    ENDIF.  
  
\*& Supply Type  
                    READ TABLE gt\_supply\_values INTO ls\_values WITH KEY domvalue\_l = lv\_supply\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-supply\_type.  
                    ENDIF.  
  
\*& Sub Supply Type  
                    READ TABLE gt\_sub\_supply\_values INTO ls\_values WITH KEY domvalue\_l = lv\_sub\_supply\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-sub\_supply\_type.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-sub\_supply\_description.  
                    ENDIF.  
  
\*& Document Type  
                    READ TABLE gt\_doctyp\_values INTO ls\_values WITH KEY domvalue\_l = lv\_document\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-document\_type.  
                    ENDIF.  
  
                    CONCATENATE '"' gw\_final-gjahr '"' INTO lw\_eway\_api\_hdr-doc\_year.  
  
                    CASE lv\_document\_number.  
                      WHEN 1.   "Billing Number as same as ODN  
                        CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-vbeln '"' INTO lw\_eway\_api\_hdr-document\_number.  
                      WHEN 2.   "ODN  
                        CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-xblnr '"' INTO lw\_eway\_api\_hdr-document\_number.  
                    ENDCASE.  
                    lw\_eway\_api\_hdr-vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln.  
  
                    CONCATENATE '"'  
                                lw\_wb2\_v\_vbrk\_vbrp2-fkdat+6(2) '/'  
                                lw\_wb2\_v\_vbrk\_vbrp2-fkdat+4(2) '/'  
                                lw\_wb2\_v\_vbrk\_vbrp2-fkdat+0(4)  
                                '"' INTO lw\_eway\_api\_hdr-document\_date.  
  
\*\*\* Supplier details  
                    CLEAR: lw\_t001w. ", lw\_adrc\_s, lw\_adr6\_s.  
                    READ TABLE gt\_t001w INTO lw\_t001w WITH  KEY werks = lw\_wb2\_v\_vbrk\_vbrp2-werks\_i BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
  
                      READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_esupgstin.  
                      IF sy-subrc IS INITIAL.  
                        CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignor.  
                      ELSE.  
                        CONCATENATE '"' gw\_final-sup\_gstin '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignor.  
                      ENDIF.  
  
                      CONCATENATE '"' lw\_t001w-name1 '"' INTO lw\_eway\_api\_hdr-legal\_name\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-name2 '"' INTO lw\_eway\_api\_hdr-address1\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-stras '"' INTO lw\_eway\_api\_hdr-address2\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-ort01 '"' INTO lw\_eway\_api\_hdr-place\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-pstlz '"' INTO lw\_eway\_api\_hdr-pincode\_of\_consignor.  
                      CLEAR:lw\_t005u\_s.  
                      READ TABLE gt\_t005u\_s INTO lw\_t005u\_s WITH  KEY bland = lw\_t001w-regio.  
                      IF sy-subrc EQ 0.  
                        TRANSLATE lw\_t005u\_s-bezei TO UPPER CASE.  
                        CONCATENATE '"' lw\_t005u\_s-bezei  '"' INTO  lw\_eway\_api\_hdr-state\_of\_consignor.  
                      ENDIF.  
  
                      CONCATENATE '"' lw\_t005u\_s-bezei '"' INTO lw\_eway\_api\_hdr-actual\_from\_state\_name.  
                    ENDIF.  
  
\*\*\* Buyer details  
                    CLEAR:lw\_kna1, lw\_adrc, lw\_adr6.  
  
                    READ TABLE gt\_vbpa INTO lw\_vbpa WITH  KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                      parvw = gc\_parvw\_re  
                                                      BINARY SEARCH.  
                    IF sy-subrc IS INITIAL.  
                      READ TABLE gt\_kna1 INTO lw\_kna1 WITH  KEY kunnr = lw\_vbpa-kunnr BINARY SEARCH.  
                      IF sy-subrc EQ 0.  
  
                        READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_ebuygstin.  
                        IF sy-subrc IS INITIAL.  
                          CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignee.  
                        ELSE.  
                          IF lw\_kna1-stcd3 IS NOT INITIAL.  
                            CONCATENATE '"' lw\_kna1-stcd3 '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignee.  
                          ELSE.  
                            lw\_eway\_api\_hdr-gstin\_of\_consignee = gc\_gstin\_urp.  
                          ENDIF.  
                        ENDIF.  
                        CONCATENATE '"' lw\_kna1-name1 '"' INTO lw\_eway\_api\_hdr-legal\_name\_of\_consignee.  
  
                        CLEAR:lw\_t005u.  
                        READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_kna1-regio.  
                        IF sy-subrc EQ 0.  
                          TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
  
                          CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_eway\_api\_hdr-actual\_to\_state\_name.  
                        ENDIF.  
  
                        CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_eway\_api\_hdr-state\_of\_supply.  
  
                        READ TABLE gt\_vbpa INTO lw\_vbpa WITH  KEY vbeln = lw\_wb2\_v\_vbrk\_vbrp2-vbeln  
                                                       parvw = gc\_parvw\_we  
                                                       BINARY SEARCH.  
                        IF sy-subrc IS INITIAL.  
                          READ TABLE gt\_kna1 INTO lw\_kna1 WITH KEY kunnr = lw\_vbpa-kunnr BINARY SEARCH.  
                          IF sy-subrc IS INITIAL.  
                            READ TABLE gt\_adrc INTO lw\_adrc WITH  KEY addrnumber = lw\_kna1-adrnr.  
                            READ TABLE gt\_adr6 INTO lw\_adr6 WITH  KEY addrnumber = lw\_adrc-addrnumber.  
  
                            CONCATENATE '"' lw\_kna1-stras '"' INTO lw\_eway\_api\_hdr-address1\_of\_consignee.  
                            CONCATENATE '"' lw\_kna1-name2 '"' INTO lw\_eway\_api\_hdr-address2\_of\_consignee.  
                            CONCATENATE '"' lw\_kna1-ort01 '"' INTO lw\_eway\_api\_hdr-place\_of\_consignee.  
\*                            CONCATENATE '"' lw\_kna1-pstlz '"' INTO lw\_eway\_api\_hdr-pincode\_of\_consignee. "Commented by Lakshmi  
                            lw\_eway\_api\_hdr-pincode\_of\_consignee = lw\_kna1-pstlz .  
                            CLEAR:lw\_t005u.  
                            READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_kna1-regio.  
                            IF sy-subrc EQ 0.  
                              TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
  
                              CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_eway\_api\_hdr-actual\_to\_state\_name.  
                            ENDIF.  
  
\* Set the export flag  
                            CLEAR lv\_export.  
                            IF lw\_t001w-land1 NE lw\_kna1-land1.  
                              lv\_export = abap\_true.  
                            ENDIF.  
  
  
                            IF lw\_eway\_api\_hdr-sub\_supply\_type      = gc\_sub\_suptyp\_exp OR  
                               lv\_export = abap\_true.  
                              lw\_eway\_api\_hdr-state\_of\_supply       = gc\_export\_place.  
                              lw\_eway\_api\_hdr-gstin\_of\_consignee    = gc\_gstin\_urp.  
                              READ TABLE gt\_export INTO ls\_export WITH KEY zport\_code = gw\_final-zport\_code.  
                              IF sy-subrc IS INITIAL..  
                                CONCATENATE '"' ls\_export-zport\_address1 '"' INTO lw\_eway\_api\_hdr-address1\_of\_consignee.  
                                CONCATENATE '"' ls\_export-zport\_address2 '"' INTO lw\_eway\_api\_hdr-address2\_of\_consignee.  
                                CONCATENATE '"' ls\_export-zport\_place '"' INTO lw\_eway\_api\_hdr-place\_of\_consignee.  
\*                                CONCATENATE '"' ls\_export-zport\_pincode '"' INTO lw\_eway\_api\_hdr-pincode\_of\_consignee. "Commented by Lakshmi  
                                lw\_eway\_api\_hdr-pincode\_of\_consignee = ls\_export-zport\_pincode.  
                                CLEAR:lw\_t005u.  
                                READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = ls\_export-zport\_state.  
                                IF sy-subrc EQ 0.  
                                  TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                                  CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_eway\_api\_hdr-actual\_to\_state\_name.  
                                ENDIF.  
                              ENDIF.  
                            ENDIF.  
                          ENDIF.  
                        ENDIF.  
                      ENDIF.  
                    ENDIF.  
  
\*\*\* Value details  
  
                    "default value  
                    lw\_eway\_api\_hdr-other\_value            = '0'.  
                    lw\_eway\_api\_hdr-total\_invoice\_value    = '0'.  
                    lw\_eway\_api\_hdr-taxable\_amount         = '0'.  
                    lw\_eway\_api\_hdr-cgst\_amount            = '0'.  
                    lw\_eway\_api\_hdr-sgst\_amount            = '0'.  
                    lw\_eway\_api\_hdr-igst\_amount            = '0'.  
                    lw\_eway\_api\_hdr-cess\_amount            = '0'.  
                    lw\_eway\_api\_hdr-cess\_nonadvol\_value    = '0'.  
  
  
\* Transaction Type  
  
                    IF ( ( lw\_eway\_api\_hdr-state\_of\_consignor = lw\_eway\_api\_hdr-actual\_from\_state\_name ) AND  
                       ( lw\_eway\_api\_hdr-state\_of\_supply = lw\_eway\_api\_hdr-actual\_to\_state\_name ) ).  
                      lw\_eway\_api\_hdr-transaction\_type = 1.  
                    ELSEIF ( ( lw\_eway\_api\_hdr-state\_of\_consignor = lw\_eway\_api\_hdr-actual\_from\_state\_name ) AND  
                       ( lw\_eway\_api\_hdr-state\_of\_supply NE lw\_eway\_api\_hdr-actual\_to\_state\_name ) ).  
                      lw\_eway\_api\_hdr-transaction\_type = 2.  
                    ELSEIF ( ( lw\_eway\_api\_hdr-state\_of\_consignor NE lw\_eway\_api\_hdr-actual\_from\_state\_name ) AND  
                       ( lw\_eway\_api\_hdr-state\_of\_supply = lw\_eway\_api\_hdr-actual\_to\_state\_name ) ).  
                      lw\_eway\_api\_hdr-transaction\_type = 3.  
                    ELSEIF ( ( lw\_eway\_api\_hdr-state\_of\_consignor NE lw\_eway\_api\_hdr-actual\_from\_state\_name ) AND  
                      ( lw\_eway\_api\_hdr-state\_of\_supply NE lw\_eway\_api\_hdr-actual\_to\_state\_name ) ).  
                      lw\_eway\_api\_hdr-transaction\_type = 4.  
                    ENDIF.  
                    CONDENSE lw\_eway\_api\_hdr-transaction\_type.  
  
  
  
                    lw\_eway\_api\_hdr-generate\_status = gc\_gen\_sts\_1.  
                    lw\_eway\_api\_hdr-data\_source = gc\_dsource\_erp.  
                    CONCATENATE '"' sy-uname '"' INTO lw\_eway\_api\_hdr-user\_ref.  
                    CONCATENATE '"' lw\_kna1-ort01 '"' INTO lw\_eway\_api\_hdr-location\_code.  
  
\*& Generate E-Way Bill Part A & Part B.  
                    IF gv\_e\_comm EQ gc\_ucomm\_eeayg.  
                      lw\_eway\_api\_hdr-eway\_bill\_status = gc\_ewaysts\_abc.  
                    ELSE.  
\*& Generate E-Way Bill Part A only.  
                      lw\_eway\_api\_hdr-eway\_bill\_status = gc\_ewaysts\_ac.  
                    ENDIF.  
  
\*& Auto print E-way Bill  
                    lw\_eway\_api\_hdr-auto\_print = gc\_eway\_print.  
                    CONCATENATE '"' lw\_adr6-smtp\_addr '"' INTO lw\_eway\_api\_hdr-email.  
  
\*\*\* Transport details  
  
                    CONCATENATE '"' gw\_final-t\_id '"' INTO lw\_eway\_api\_hdr-transporter\_id.  
                    CONCATENATE '"' gw\_final-t\_name '"' INTO lw\_eway\_api\_hdr-transporter\_name.  
  
                    READ TABLE gt\_t\_mode\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-t\_mode.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-transportation\_mode.  
                    ELSE.  
                      CONCATENATE '"' gw\_final-t\_mode '"' INTO lw\_eway\_api\_hdr-transportation\_mode.  
                    ENDIF.  
                    CONCATENATE '"' gw\_final-t\_distance '"' INTO lw\_eway\_api\_hdr-transportation\_distance.  
  
                    CONCATENATE '"' gw\_final-t\_doc\_no '"' INTO lw\_eway\_api\_hdr-transporter\_document\_number.  
                    CONCATENATE '"' gw\_final-t\_date '"' INTO lw\_eway\_api\_hdr-transporter\_document\_date.  
  
                    CONCATENATE '"' gw\_final-v\_number '"' INTO lw\_eway\_api\_hdr-vehicle\_number.  
                    CLEAR ls\_values.  
                    READ TABLE gt\_v\_type\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-v\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
  
                    ENDIF.  
                    CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-vehicle\_type.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill item derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
\* check for export currency  
                    CLEAR lv\_exp\_curr.  
                    IF lw\_wb2\_v\_vbrk\_vbrp2-waerk NE  gc\_curr\_inr .  
                      lv\_exp\_curr = abap\_true.  
                    ENDIF.  
  
  
\*\*\* Rest item tax rate  
                    lw\_eway\_api\_itm-cgst\_rate = '0'.  
                    lw\_eway\_api\_itm-sgst\_rate = '0'.  
                    lw\_eway\_api\_itm-igst\_rate = '0'.  
                    lw\_eway\_api\_itm-cess\_rate = '0'.  
                    lw\_eway\_api\_itm-cessnonadvol = '0'.  
  
  
                    lw\_eway\_api\_itm-bukrs            = lw\_eway\_api\_hdr-bukrs.  
                    lw\_eway\_api\_itm-document\_number  = lw\_eway\_api\_hdr-document\_number.  
                    lw\_eway\_api\_itm-doc\_year         = lw\_eway\_api\_hdr-doc\_year.  
                    lw\_eway\_api\_itm-document\_type    = lw\_eway\_api\_hdr-document\_type.  
                    lw\_eway\_api\_itm-document\_date    = lw\_eway\_api\_hdr-document\_date.  
  
                    CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i '"' INTO lw\_eway\_api\_itm-product\_name.  
                    CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-arktx\_i '"' INTO lw\_eway\_api\_itm-product\_description.  
  
                    READ TABLE gt\_marc INTO lw\_marc WITH KEY matnr = lw\_wb2\_v\_vbrk\_vbrp2-matnr\_i  
                                                             werks = lw\_wb2\_v\_vbrk\_vbrp2-werks\_i  
                                                             BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"' lw\_marc-steuc '"' INTO lw\_eway\_api\_itm-hsn\_code.  
                    ENDIF.  
  
                    lw\_eway\_api\_itm-quantity         = lw\_wb2\_v\_vbrk\_vbrp2-fkimg\_i.  
  
                    CLEAR lv\_output\_uom.  
                    CALL FUNCTION 'CONVERSION\_EXIT\_CUNIT\_OUTPUT'  
                      EXPORTING  
                        input          = lw\_wb2\_v\_vbrk\_vbrp2-meins\_i  
                      IMPORTING  
                        output         = lv\_output\_uom  
                      EXCEPTIONS  
                        unit\_not\_found = 1  
                        OTHERS         = 2.  
                    IF sy-subrc <> 0.  
\* Implement suitable error handling here  
                    ENDIF.  
  
                    READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins = lv\_output\_uom.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_eway\_api\_itm-unit\_of\_product.  
                    ELSE.  
  
                      READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins =  lw\_wb2\_v\_vbrk\_vbrp2-meins\_i.  
                      IF sy-subrc IS INITIAL.  
                        CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_eway\_api\_itm-unit\_of\_product.  
                      ELSE.  
                        CONCATENATE '"' lw\_wb2\_v\_vbrk\_vbrp2-meins\_i '"' INTO  lw\_eway\_api\_itm-unit\_of\_product.  
  
                      ENDIF.  
                    ENDIF.  
                    "Item Amount  
  
                    lw\_eway\_api\_itm-taxable\_amount = lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i.  
                    IF lv\_exp\_curr IS NOT INITIAL.  
                      PERFORM convert\_currency USING lw\_eway\_api\_itm-taxable\_amount lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                             CHANGING lw\_eway\_api\_itm-taxable\_amount.  
                    ENDIF.  
  
                    "CGST Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_jocg.  
                    IF sy-subrc EQ 0.  
                      IF lv\_exp\_curr IS NOT INITIAL.  
                        PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                               CHANGING lw\_konv-kwert.  
                      ENDIF.  
                      lw\_eway\_api\_itm-cgst\_rate = lw\_konv-kbetr.  
                      lv\_total\_cgst\_value = lv\_total\_cgst\_value + lw\_konv-kwert.  
                    ENDIF.  
                    "SGST Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_josg.  
                    IF sy-subrc IS NOT INITIAL.  
                      READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                               kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                               kschl = gc\_kschl\_joug.  
                    ENDIF.  
                    IF sy-subrc EQ 0.  
                      IF lv\_exp\_curr IS NOT INITIAL.  
                        PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                               CHANGING lw\_konv-kwert.  
                      ENDIF.  
                      lw\_eway\_api\_itm-sgst\_rate = lw\_konv-kbetr.  
                      lv\_total\_sgst\_value = lv\_total\_sgst\_value + lw\_konv-kwert.  
                    ENDIF.  
                    "TCS amount  
\*                     Code begin added by Lakshmi  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                               kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                               kschl = gc\_kschl\_jwts  
                                                               BINARY SEARCH.  
                    IF sy-subrc = 0.  
                      lw\_eway\_api\_hdr-other\_value =  lw\_eway\_api\_hdr-other\_value + lw\_konv-kwert.  
                    ENDIF.  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                               kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                               kschl = gc\_kschl\_jtc1  
                                                               BINARY SEARCH.  
                    IF sy-subrc = 0.  
                      lw\_eway\_api\_hdr-other\_value =  lw\_eway\_api\_hdr-other\_value + lw\_konv-kwert.  
                    ENDIF.  
                    CONDENSE  lw\_eway\_api\_hdr-other\_value.  
\*                  Code ended added by Lakshmi  
                    "IGST Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_joig.  
                    IF sy-subrc EQ 0.  
                      IF lv\_exp\_curr IS NOT INITIAL.  
                        PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                               CHANGING lw\_konv-kwert.  
                      ENDIF.  
                      lw\_eway\_api\_itm-igst\_rate = lw\_konv-kbetr.  
                      lv\_total\_igst\_value = lv\_total\_igst\_value + lw\_konv-kwert.  
                    ENDIF.  
                    "CESS Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                             kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                             kschl = gc\_kschl\_cess.  
                    IF sy-subrc EQ 0.  
                      IF lv\_exp\_curr IS NOT INITIAL.  
                        PERFORM convert\_currency USING lw\_konv-kwert lw\_wb2\_v\_vbrk\_vbrp2-kursk\_i  
                                               CHANGING lw\_konv-kwert.  
                      ENDIF.  
                      lw\_eway\_api\_itm-cess\_rate = lw\_konv-kbetr.  
                      lv\_total\_cess\_value = lv\_total\_cess\_value + lw\_konv-kwert.  
                    ENDIF.  
                    lw\_eway\_api\_itm-cessnonadvol = 0.  
                    lv\_total\_cessnoval\_value = lv\_total\_cessnoval\_value + 0.  
  
  
                    lv\_total\_taxable\_value  = lv\_total\_taxable\_value  + lw\_wb2\_v\_vbrk\_vbrp2-netwr\_i.  
                    APPEND lw\_eway\_api\_itm TO gt\_eway\_api\_itm.  
                    CLEAR:lw\_eway\_api\_itm.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  End of Fill item derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
                  ENDLOOP.  
  
  
\*\*\*\* Header total values  
  
                  lw\_eway\_api\_hdr-total\_invoice\_value  = lv\_total\_taxable\_value +  
                                                         lv\_total\_cgst\_value +  
                                                         lv\_total\_sgst\_value +  
                                                         lv\_total\_igst\_value +  
                                                         lv\_total\_cess\_value +  
                                                         lv\_total\_cessnoval\_value +  
                                                         lw\_eway\_api\_hdr-other\_value.  
  
\*                  lw\_eway\_api\_hdr-other\_value          = 0.  
                  lw\_eway\_api\_hdr-taxable\_amount       = lv\_total\_taxable\_value.  
                  lw\_eway\_api\_hdr-cgst\_amount          = lv\_total\_cgst\_value.  
                  lw\_eway\_api\_hdr-sgst\_amount          = lv\_total\_sgst\_value.  
                  lw\_eway\_api\_hdr-igst\_amount          = lv\_total\_igst\_value.  
                  lw\_eway\_api\_hdr-cess\_amount          = lv\_total\_cess\_value.  
                  lw\_eway\_api\_hdr-cess\_nonadvol\_value  = lv\_total\_cessnoval\_value.  
  
                  APPEND lw\_eway\_api\_hdr TO gt\_eway\_api\_hdr.  
                  CLEAR:lw\_eway\_api\_hdr.  
  
                ELSE.  
  
                  MESSAGE 'Please select appropriate documents for E-Doc generation'(039) TYPE 'I'.  
                ENDIF.  
\*\*\*\* Header total values  
\* Delivery Challan  
              ELSE.  
  
                READ TABLE gt\_wb2\_v\_likp\_lips2 INTO lw\_wb2\_v\_likp\_lips2 WITH KEY vbeln = gw\_final-vbeln.  
                IF sy-subrc = 0.  
  
  
                  lv\_index = sy-tabix.  
  
\*\*\* Reset header value for every document  
                  lv\_total\_invoice\_value   = 0.  
                  lv\_total\_sgst\_value      = 0.  
                  lv\_total\_cgst\_value      = 0.  
                  lv\_total\_igst\_value      = 0.  
                  lv\_total\_cess\_value      = 0.  
                  lv\_total\_cessnoval\_value = 0.  
                  lv\_total\_taxable\_value   = 0.  
  
                  LOOP AT gt\_wb2\_v\_likp\_lips2 INTO lw\_wb2\_v\_likp\_lips2 FROM lv\_index.  
                    IF lw\_wb2\_v\_likp\_lips2-vbeln NE gw\_final-vbeln.  
                      EXIT.  
                    ENDIF.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
                    lw\_eway\_api\_hdr-bukrs       = p\_ccode.  
  
                    READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_eusrgstin.  
                    IF sy-subrc IS INITIAL.  
                      CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO lw\_eway\_api\_hdr-user\_gstin.  
                    ELSE.  
                      CONCATENATE '"' gw\_final-sup\_gstin '"' INTO lw\_eway\_api\_hdr-user\_gstin.  
                    ENDIF.  
  
                    READ TABLE gt\_doctyp INTO ls\_doctyp WITH KEY fkart = lw\_wb2\_v\_likp\_lips2-lfart.  
                    IF sy-subrc EQ 0.  
                      lv\_supply\_type = ls\_doctyp-sup\_type.  
                      lv\_sub\_supply\_type = ls\_doctyp-sub\_type.  
                      lv\_document\_type = ls\_doctyp-edoc\_type.  
                    ENDIF.  
  
\*& Supply Type  
                    READ TABLE gt\_supply\_values INTO ls\_values WITH KEY domvalue\_l = lv\_supply\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-supply\_type.  
                    ENDIF.  
  
\*& Sub Supply Type  
                    READ TABLE gt\_sub\_supply\_values INTO ls\_values WITH KEY domvalue\_l = lv\_sub\_supply\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-sub\_supply\_type.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-sub\_supply\_description.  
                    ENDIF.  
  
\*& Document Type  
                    READ TABLE gt\_doctyp\_values INTO ls\_values WITH KEY domvalue\_l = lv\_document\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-document\_type.  
                    ENDIF.  
  
                    CONCATENATE '"' lw\_wb2\_v\_likp\_lips2-lfgja\_i '"' INTO lw\_eway\_api\_hdr-doc\_year.  
  
                    CASE lv\_document\_number.  
                      WHEN 1.   "Billing Number as same as ODN  
                        CONCATENATE '"' lw\_wb2\_v\_likp\_lips2-vbeln '"' INTO lw\_eway\_api\_hdr-document\_number.  
                      WHEN 2.   "ODN  
                        CONCATENATE '"' lw\_wb2\_v\_likp\_lips2-xblnr '"' INTO lw\_eway\_api\_hdr-document\_number.  
                    ENDCASE.  
                    lw\_eway\_api\_hdr-vbeln = lw\_wb2\_v\_likp\_lips2-vbeln.  
  
                    CONCATENATE '"'  
                                lw\_wb2\_v\_likp\_lips2-fkdat+6(2) '/'  
                                lw\_wb2\_v\_likp\_lips2-fkdat+4(2) '/'  
                                lw\_wb2\_v\_likp\_lips2-fkdat+0(4)  
                                '"' INTO lw\_eway\_api\_hdr-document\_date.  
  
\*\*\* Supplier details  
                    CLEAR: lw\_t001w.  
                    READ TABLE gt\_t001w INTO lw\_t001w WITH  KEY werks = lw\_wb2\_v\_likp\_lips2-werks\_i BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
  
                      READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_esupgstin.  
                      IF sy-subrc IS INITIAL.  
                        CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignor.  
                      ELSE.  
                        CONCATENATE '"' gw\_final-sup\_gstin '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignor.  
                      ENDIF.  
  
  
                      CONCATENATE '"' lw\_t001w-name1 '"' INTO lw\_eway\_api\_hdr-legal\_name\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-name2 '"' INTO lw\_eway\_api\_hdr-address1\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-stras '"' INTO lw\_eway\_api\_hdr-address2\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-ort01 '"' INTO lw\_eway\_api\_hdr-place\_of\_consignor.  
                      CONCATENATE '"' lw\_t001w-pstlz '"' INTO lw\_eway\_api\_hdr-pincode\_of\_consignor.  
                      CLEAR:lw\_t005u\_s.  
                      READ TABLE gt\_t005u\_s INTO lw\_t005u\_s WITH  KEY bland = lw\_t001w-regio.  
                      IF sy-subrc EQ 0.  
                        TRANSLATE lw\_t005u\_s-bezei TO UPPER CASE.  
                        CONCATENATE '"' lw\_t005u\_s-bezei  '"' INTO  lw\_eway\_api\_hdr-state\_of\_consignor.  
                      ENDIF.  
  
                      CONCATENATE '"' lw\_t005u\_s-bezei '"' INTO lw\_eway\_api\_hdr-actual\_from\_state\_name.  
  
                    ELSE.  
                    ENDIF.  
  
\*\*\* Buyer details  
                    CLEAR:lw\_kna1, lw\_adrc, lw\_adr6.  
                    READ TABLE gt\_kna1 INTO lw\_kna1 WITH  KEY kunnr = lw\_wb2\_v\_likp\_lips2-kunag BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
  
                      READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_ebuygstin.  
                      IF sy-subrc IS INITIAL.  
                        CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignee.  
                      ELSE.  
                        CONCATENATE '"' lw\_kna1-stcd3 '"' INTO lw\_eway\_api\_hdr-gstin\_of\_consignee.  
                      ENDIF.  
  
                      READ TABLE gt\_adrc INTO lw\_adrc WITH  KEY addrnumber = lw\_kna1-adrnr.  
                      READ TABLE gt\_adr6 INTO lw\_adr6 WITH  KEY addrnumber = lw\_adrc-addrnumber.  
  
                      CONCATENATE '"' lw\_kna1-name1 '"' INTO lw\_eway\_api\_hdr-legal\_name\_of\_consignee.  
                      CONCATENATE '"' lw\_kna1-stras '"' INTO lw\_eway\_api\_hdr-address1\_of\_consignee.  
                      CONCATENATE '"' lw\_kna1-name2 '"' INTO lw\_eway\_api\_hdr-address2\_of\_consignee.  
                      CONCATENATE '"' lw\_kna1-ort01 '"' INTO lw\_eway\_api\_hdr-place\_of\_consignee.  
\*                      CONCATENATE '"' lw\_t001w-pstlz '"' INTO lw\_eway\_api\_hdr-pincode\_of\_consignee."Commented by Lakshmi  
                      lw\_eway\_api\_hdr-pincode\_of\_consignee = lw\_t001w-pstlz.  
                      CLEAR:lw\_t005u.  
                      READ TABLE gt\_t005u INTO lw\_t005u WITH  KEY bland = lw\_kna1-regio.  
                      IF sy-subrc EQ 0.  
                        TRANSLATE lw\_t005u-bezei TO UPPER CASE.  
                        CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_eway\_api\_hdr-state\_of\_supply.  
                      ENDIF.  
                      CONCATENATE '"' lw\_t005u-bezei '"' INTO lw\_eway\_api\_hdr-actual\_to\_state\_name.  
  
                    ENDIF.  
  
\*\*\* Value details  
  
                    "default value  
                    lw\_eway\_api\_hdr-other\_value            = '0'.  
                    lw\_eway\_api\_hdr-total\_invoice\_value    = '0'.  
                    lw\_eway\_api\_hdr-taxable\_amount         = '0'.  
                    lw\_eway\_api\_hdr-cgst\_amount            = '0'.  
                    lw\_eway\_api\_hdr-sgst\_amount            = '0'.  
                    lw\_eway\_api\_hdr-igst\_amount            = '0'.  
                    lw\_eway\_api\_hdr-cess\_amount            = '0'.  
                    lw\_eway\_api\_hdr-cess\_nonadvol\_value    = '0'.  
  
  
                    CONCATENATE '"' 'REGULAR' '"' INTO lw\_eway\_api\_hdr-transaction\_type.  
                    CONCATENATE '"' '1' '"' INTO lw\_eway\_api\_hdr-generate\_status.  
                    CONCATENATE '"' 'ERP' '"' INTO lw\_eway\_api\_hdr-data\_source.  
                    CONCATENATE '"' sy-uname '"' INTO lw\_eway\_api\_hdr-user\_ref.  
                    CONCATENATE '"' lw\_kna1-ort01 '"' INTO lw\_eway\_api\_hdr-location\_code.  
  
\*& Generate E-Way Bill Part A & Part B.  
                    IF gv\_e\_comm EQ 'EEAYG'.  
                      CONCATENATE '"' 'ABC' '"' INTO lw\_eway\_api\_hdr-eway\_bill\_status.  
                    ELSE.  
\*& Generate E-Way Bill Part A only.  
                      CONCATENATE '"' 'AC' '"' INTO lw\_eway\_api\_hdr-eway\_bill\_status.  
                    ENDIF.  
  
\*& Auto print E-way Bill  
                    CONCATENATE '"' 'Y' '"' INTO lw\_eway\_api\_hdr-auto\_print.  
                    CONCATENATE '"' lw\_adr6-smtp\_addr '"' INTO lw\_eway\_api\_hdr-email.  
  
\*\*\* Transport details  
  
  
                    CONCATENATE '"' gw\_final-t\_id '"' INTO lw\_eway\_api\_hdr-transporter\_id.  
                    CONCATENATE '"' gw\_final-t\_name '"' INTO lw\_eway\_api\_hdr-transporter\_name.  
  
                    READ TABLE gt\_t\_mode\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-t\_mode.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
                      CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-transportation\_mode.  
                    ENDIF.  
                    CONCATENATE '"' gw\_final-t\_distance '"' INTO lw\_eway\_api\_hdr-transportation\_distance.  
  
                    CONCATENATE '"' gw\_final-t\_doc\_no '"' INTO lw\_eway\_api\_hdr-transporter\_document\_number.  
                    CONCATENATE '"' gw\_final-t\_date '"' INTO lw\_eway\_api\_hdr-transporter\_document\_date.  
  
                    CONCATENATE '"' gw\_final-v\_number '"' INTO lw\_eway\_api\_hdr-vehicle\_number.  
                    CLEAR ls\_values.  
                    READ TABLE gt\_v\_type\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-v\_type.  
                    IF sy-subrc EQ 0.  
                      TRANSLATE ls\_values-ddtext TO UPPER CASE.  
  
  
                    ENDIF.  
                    CONCATENATE '"' ls\_values-ddtext '"' INTO lw\_eway\_api\_hdr-vehicle\_type.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill item derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\* Rest item tax rate  
                    lw\_eway\_api\_itm-cgst\_rate = '0'.  
                    lw\_eway\_api\_itm-sgst\_rate = '0'.  
                    lw\_eway\_api\_itm-igst\_rate = '0'.  
                    lw\_eway\_api\_itm-cess\_rate = '0'.  
                    lw\_eway\_api\_itm-cessnonadvol = '0'.  
  
  
                    lw\_eway\_api\_itm-bukrs            = lw\_eway\_api\_hdr-bukrs.  
                    lw\_eway\_api\_itm-document\_number  = lw\_eway\_api\_hdr-document\_number.  
                    lw\_eway\_api\_itm-doc\_year         = lw\_eway\_api\_hdr-doc\_year.  
                    lw\_eway\_api\_itm-document\_type    = lw\_eway\_api\_hdr-document\_type.  
                    lw\_eway\_api\_itm-document\_date    = lw\_eway\_api\_hdr-document\_date.  
  
                    CONCATENATE '"' lw\_wb2\_v\_likp\_lips2-matnr\_i '"' INTO lw\_eway\_api\_itm-product\_name.  
                    CONCATENATE '"' lw\_wb2\_v\_likp\_lips2-arktx\_i '"' INTO lw\_eway\_api\_itm-product\_description.  
  
                    READ TABLE gt\_marc INTO lw\_marc WITH KEY matnr = lw\_wb2\_v\_likp\_lips2-matnr\_i  
                                                             werks = lw\_wb2\_v\_likp\_lips2-werks\_i  
                                                             BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"' lw\_marc-steuc '"' INTO lw\_eway\_api\_itm-hsn\_code.  
                    ENDIF.  
  
                    lw\_eway\_api\_itm-quantity         = lw\_wb2\_v\_likp\_lips2-lfimg\_i.  
  
                    CLEAR lv\_output\_uom.  
                    CALL FUNCTION 'CONVERSION\_EXIT\_CUNIT\_OUTPUT'  
                      EXPORTING  
                        input          = lw\_wb2\_v\_likp\_lips2-meins\_i  
                      IMPORTING  
                        output         = lv\_output\_uom  
                      EXCEPTIONS  
                        unit\_not\_found = 1  
                        OTHERS         = 2.  
                    IF sy-subrc <> 0.  
\* Implement suitable error handling here  
                    ENDIF.  
  
                    READ TABLE gt\_doc\_uom INTO ls\_doc\_uom WITH KEY meins = lv\_output\_uom.  
                    IF sy-subrc EQ 0.  
                      CONCATENATE '"' ls\_doc\_uom-edoc\_uom '"' INTO lw\_eway\_api\_itm-unit\_of\_product.  
                    ENDIF.  
                    "Item Amount  
                    lw\_eway\_api\_itm-taxable\_amount = lw\_wb2\_v\_likp\_lips2-netwr\_i.  
  
                    "CGST Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_likp\_lips2-knumv  
                                                             kposn = lw\_wb2\_v\_likp\_lips2-posnr\_i  
                                                             kschl = gc\_kschl\_jocg  
                                                             BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      lw\_eway\_api\_itm-cgst\_rate = lw\_konv-kbetr.  
                      lv\_total\_cgst\_value = lv\_total\_cgst\_value + lw\_konv-kwert.  
                    ENDIF.  
                    "SGST Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_likp\_lips2-knumv  
                                                             kposn = lw\_wb2\_v\_likp\_lips2-posnr\_i  
                                                             kschl = gc\_kschl\_josg  
                                                             BINARY SEARCH.  
                    IF sy-subrc IS NOT INITIAL.  
                      READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_vbrk\_vbrp2-knumv  
                                                              kposn = lw\_wb2\_v\_vbrk\_vbrp2-posnr\_i  
                                                              kschl = gc\_kschl\_joug.  
                    ENDIF.  
                    IF sy-subrc EQ 0.  
                      lw\_eway\_api\_itm-sgst\_rate = lw\_konv-kbetr.  
                      lv\_total\_sgst\_value = lv\_total\_sgst\_value + lw\_konv-kwert.  
                    ENDIF.  
                    "IGST Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_likp\_lips2-knumv  
                                                             kposn = lw\_wb2\_v\_likp\_lips2-posnr\_i  
                                                             kschl = gc\_kschl\_joig  
                                                             BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      lw\_eway\_api\_itm-igst\_rate = lw\_konv-kbetr.  
                      lv\_total\_igst\_value = lv\_total\_igst\_value + lw\_konv-kwert.  
                    ENDIF.  
                    "CESS Tax amount  
                    READ TABLE gt\_konv INTO lw\_konv WITH KEY knumv = lw\_wb2\_v\_likp\_lips2-knumv  
                                                             kposn = lw\_wb2\_v\_likp\_lips2-posnr\_i  
                                                             kschl = gc\_kschl\_cess  
                                                             BINARY SEARCH.  
                    IF sy-subrc EQ 0.  
                      lw\_eway\_api\_itm-cess\_rate = lw\_konv-kbetr.  
                      lv\_total\_cess\_value = lv\_total\_cess\_value + lw\_konv-kwert.  
                    ENDIF.  
                    lw\_eway\_api\_itm-cessnonadvol = 0.  
                    lv\_total\_cessnoval\_value = lv\_total\_cessnoval\_value + 0.  
  
                    lv\_total\_taxable\_value = lv\_total\_taxable\_value + lw\_wb2\_v\_likp\_lips2-netwr.  
  
                    APPEND lw\_eway\_api\_itm TO gt\_eway\_api\_itm.  
                    CLEAR:lw\_eway\_api\_itm.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  End of Fill item derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
                  ENDLOOP.  
  
                  lw\_eway\_api\_hdr-total\_invoice\_value  = lv\_total\_taxable\_value +  
                                                         lv\_total\_cgst\_value +  
                                                         lv\_total\_sgst\_value +  
                                                         lv\_total\_igst\_value +  
                                                         lv\_total\_cess\_value +  
                                                         lv\_total\_cessnoval\_value.  
  
\*                  lw\_eway\_api\_hdr-other\_value          = 0.  
                  lw\_eway\_api\_hdr-taxable\_amount       = lv\_total\_taxable\_value.  
                  lw\_eway\_api\_hdr-cgst\_amount          = lv\_total\_cgst\_value.  
                  lw\_eway\_api\_hdr-sgst\_amount          = lv\_total\_sgst\_value.  
                  lw\_eway\_api\_hdr-igst\_amount          = lv\_total\_igst\_value.  
                  lw\_eway\_api\_hdr-cess\_amount          = lv\_total\_cess\_value.  
                  lw\_eway\_api\_hdr-cess\_nonadvol\_value  = lv\_total\_cessnoval\_value.  
  
                  APPEND lw\_eway\_api\_hdr TO gt\_eway\_api\_hdr.  
                  CLEAR:lw\_eway\_api\_hdr.  
  
                ENDIF.  
              ENDIF.  
            ENDIF.  
          ELSE.  
            MESSAGE 'Please select appropriate documents for E-Doc generation'(039) TYPE 'I'.  
          ENDIF.  
        ENDLOOP.  
  
  
        IF gt\_eway\_api\_hdr IS NOT INITIAL.  
          SORT gt\_eway\_api\_hdr .  
          SORT gt\_eway\_api\_itm .  
  
          DELETE ADJACENT DUPLICATES FROM   gt\_eway\_api\_hdr  COMPARING ALL FIELDS.  
          DELETE ADJACENT DUPLICATES FROM  gt\_einv\_api\_itm  COMPARING ALL FIELDS.  
  
          lt\_api\_hdr = gt\_eway\_api\_hdr .  
          lt\_api\_itm = gt\_eway\_api\_itm .  
          CLEAR lv\_error.  
          CALL FUNCTION 'ZFM\_EWAY\_BILL\_VALIDATION'  
            EXPORTING  
              im\_ucomm      = gv\_e\_comm  
              im\_api\_header = gt\_eway\_api\_hdr  
            IMPORTING  
              ex\_error      = lv\_error.  
          IF lv\_error = ' '.  
            CLEAR:lw\_token,lw\_return.  
            IF gw\_token IS NOT INITIAL.  
              lw\_token = gw\_token.  
            ELSE.  
              CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
                IMPORTING  
                  ex\_token    = lw\_token  
                  ex\_return   = lw\_return  
                  et\_messages = lt\_messages.  
            ENDIF.  
            IF lw\_token IS NOT INITIAL.  
              CALL FUNCTION 'ZFM\_EWAY\_BILL\_GENERATE\_API'  
                EXPORTING  
                  im\_token          = lw\_token  
                  im\_api\_header     = lt\_api\_hdr  
                  im\_api\_item       = lt\_api\_itm  
                IMPORTING  
                  ex\_return         = lw\_return  
                  ex\_messages       = lt\_messages  
                  ex\_ewaybill       = lt\_ewaybill  
                  ex\_eway\_transport = lt\_eway\_transport.  
  
              IF lt\_ewaybill IS NOT INITIAL.  
                LOOP AT lt\_ewaybill ASSIGNING <lw\_ewaybill>.  
                  READ TABLE gt\_final INTO gw\_final WITH KEY vbeln = <lw\_ewaybill>-docno.  
                  IF sy-subrc = 0.  
                    <lw\_ewaybill>-gjahr = gw\_final-gjahr.  
                    <lw\_ewaybill>-doctyp = gw\_final-fkart.  
                  ENDIF.  
                ENDLOOP.  
                MODIFY j\_1ig\_ewaybill FROM TABLE lt\_ewaybill.  
                IF sy-subrc EQ 0.  
                  COMMIT WORK.  
  
                  LOOP AT lt\_ewaybill INTO lw\_ewaybill.  
                    READ TABLE gt\_final ASSIGNING <gw\_final> WITH KEY vbeln = lw\_ewaybill-docno.  
                    IF sy-subrc IS INITIAL.  
                      <gw\_final>-eway\_num = lw\_ewaybill-ebillno.  
  
                      CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                        EXPORTING  
                          date\_internal            = lw\_ewaybill-egen\_dat  
                        IMPORTING  
                          date\_external            = <gw\_final>-eway\_date  
                        EXCEPTIONS  
                          date\_internal\_is\_invalid = 1  
                          OTHERS                   = 2.  
                      IF sy-subrc <> 0.  
                        MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
                      ENDIF.  
  
                      <gw\_final>-eway\_ernam = lw\_ewaybill-ernam.  
  
                      CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                        EXPORTING  
                          date\_internal            = lw\_ewaybill-erdat  
                        IMPORTING  
                          date\_external            = <gw\_final>-eway\_erdat  
                        EXCEPTIONS  
                          date\_internal\_is\_invalid = 1  
                          OTHERS                   = 2.  
                      IF sy-subrc <> 0.  
                        MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
                      ENDIF.  
  
                      <gw\_final>-eway\_erzet = lw\_ewaybill-egen\_time.  
  
\*Valid upto  
  
  
                      CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                        EXPORTING  
                          date\_internal            = lw\_ewaybill-vdtodate  
                        IMPORTING  
                          date\_external            = <gw\_final>-eway\_v\_to  
                        EXCEPTIONS  
                          date\_internal\_is\_invalid = 1  
                          OTHERS                   = 2.  
                      IF sy-subrc <> 0.  
                        MESSAGE 'Date conversion error'(m02) TYPE 'I'.  
                      ENDIF.  
  
                      IF lw\_ewaybill-status = gc\_eway\_sts\_a.  
                        <gw\_final>-eway\_status = 'Success'(060).  
                        <gw\_final>-eway\_icon    = gc\_icon\_08.  
                        <gw\_final>-eway\_error   = space.  
                      ELSEIF lw\_ewaybill-status = gc\_eway\_sts\_e.  
                        <gw\_final>-eway\_status = 'Error'(061).  
                        <gw\_final>-eway\_icon    = gc\_icon\_0a.  
                      ELSEIF lw\_ewaybill-status = gc\_eway\_sts\_c.  
                        <gw\_final>-eway\_status = 'Cancelled'(062).  
                        <gw\_final>-eway\_icon    = gc\_icon\_0w.  
                      ENDIF.  
                    ENDIF.  
                  ENDLOOP.  
  
                  CLEAR lt\_show\_message.  
                  IF  lt\_messages IS NOT INITIAL.  
                    LOOP AT gt\_index INTO lw\_index.  
  
                      READ TABLE gt\_final ASSIGNING <gw\_final> INDEX lw\_index-index.  
                      IF sy-subrc EQ 0.  
  
                        CLEAR: lv\_string,lw\_etransport.  
                        LOOP AT lt\_messages INTO lw\_message WHERE message\_v3 = <gw\_final>-vbeln.  
  
                          lw\_show\_message-msgid =  gc\_msgid\_01.  
                          lw\_show\_message-msgty =  lw\_message-type.  
                          lw\_show\_message-msgno =  gc\_msgno\_319.  
                          lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
                          lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
                          lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
                          lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
                          APPEND lw\_show\_message TO lt\_show\_message.  
  
                          IF lw\_message-type = gc\_mtype\_s.  
                            READ TABLE lt\_eway\_transport INTO ls\_eway\_transport WITH KEY docno = <gw\_final>-vbeln.  
                            IF sy-subrc IS INITIAL.  
                              <gw\_final>-eway\_print = ls\_eway\_transport-eway\_print.  
                            ELSE.  
                              <gw\_final>-eway\_print = lw\_message-message\_v4.  
                            ENDIF.  
                            UPDATE zteway\_transport SET eway\_print = <gw\_final>-eway\_print  
                                                        WHERE  bukrs = <gw\_final>-bukrs AND  
                                                               gjahr = <gw\_final>-gjahr AND  
                                                               docno = lw\_message-message\_v3 AND  
                                                               doctyp = <gw\_final>-fkart.  
                            IF sy-subrc IS NOT INITIAL.  
                              MOVE-CORRESPONDING <gw\_final> TO lw\_etransport.  
                              lw\_etransport-t\_date = <gw\_final>-fkdat\_db.  
                              lw\_etransport-doctyp = <gw\_final>-fkart.  
                              lw\_etransport-docno = <gw\_final>-vbeln.  
                              lw\_etransport-gjahr =  <gw\_final>-gjahr.  
  
                              READ TABLE lt\_eway\_transport INTO ls\_eway\_transport WITH KEY docno = <gw\_final>-vbeln.  
                              IF sy-subrc IS INITIAL.  
                                <gw\_final>-eway\_print = ls\_eway\_transport-eway\_print.  
  
                              ELSE.  
                                <gw\_final>-eway\_print = lw\_message-message\_v4.  
                              ENDIF.  
                              lw\_etransport-gjahr =  <gw\_final>-eway\_print.  
                              lw\_etransport-ernam = sy-uname.  
                              lw\_etransport-erdat = sy-datum.  
  
                              MODIFY zteway\_transport FROM lw\_etransport.  
                              IF sy-subrc IS NOT INITIAL.  
                                MESSAGE 'Data update error'(022) TYPE 'I'.  
                              ENDIF.  
                            ENDIF.  
                            CONTINUE.  
                          ELSEIF lw\_message-type = gc\_mtype\_e.  
  
                            CONCATENATE lv\_string lw\_message-message INTO lv\_string SEPARATED BY '||'.  
  
                          ENDIF.  
  
                        ENDLOOP.  
                        IF lv\_string IS NOT INITIAL.  
                          <gw\_final>-eway\_error = lv\_string.  
                          UPDATE zteway\_transport SET eway\_error = lv\_string  
                                                      WHERE  bukrs = <gw\_final>-bukrs AND  
                                                             gjahr = <gw\_final>-gjahr AND  
                                                             docno = lw\_message-message\_v3 AND  
                                                             doctyp = <gw\_final>-fkart.  
                          IF sy-subrc IS NOT INITIAL.  
                            MOVE-CORRESPONDING <gw\_final> TO lw\_etransport.  
                            lw\_etransport-t\_date = <gw\_final>-fkdat\_db.  
                            lw\_etransport-doctyp = <gw\_final>-fkart.  
                            lw\_etransport-docno = <gw\_final>-vbeln.  
                            lw\_etransport-ernam = sy-uname.  
                            lw\_etransport-erdat = sy-datum.  
                            MODIFY zteway\_transport FROM lw\_etransport.  
                            IF sy-subrc IS NOT INITIAL.  
                              MESSAGE 'Data update error'(022) TYPE 'I'.  
                            ENDIF.  
                          ENDIF.  
                        ENDIF.  
                      ENDIF.  
                    ENDLOOP.  
  
  
                    CALL METHOD gref\_alv\_grid->refresh\_table\_display  
                      EXCEPTIONS  
                        finished = 1  
                        OTHERS   = 2.  
                    IF sy-subrc <> 0.  
                      MESSAGE 'Table refresh error'(012) TYPE 'I'.  
                    ENDIF.  
  
  
                    CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
                      TABLES  
                        i\_message\_tab = lt\_show\_message.  
  
                  ENDIF.  
  
  
                ENDIF.  
  
              ENDIF.  
  
  
            ELSE.  
              MESSAGE 'Access token generation error'(015) TYPE 'I'.  
              LEAVE LIST-PROCESSING.  
  
            ENDIF.  
  
          ENDIF.  
        ENDIF.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
  
ENDFORM.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_UPDATE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_update\_transporter.  
  
  
  
\*  
\*  DATA:lv\_index        TYPE sy-tabix.  
\*  
  DATA:lw\_token                    TYPE string,  
       lw\_return                   TYPE string,  
       wa\_return                   TYPE string,  
       lt\_messages                 TYPE bapiret2\_t,  
       lw\_message                  TYPE bapiret2,  
       lt\_show\_message             TYPE esp1\_message\_tab\_type,  
       lw\_show\_message             LIKE LINE OF lt\_show\_message,  
       lt\_ewaybill                 TYPE TABLE OF j\_1ig\_ewaybill,  
       lw\_ewaybill                 TYPE j\_1ig\_ewaybill,  
       lw\_index                    TYPE lvc\_s\_row,  
       lv\_val                      TYPE xfeld,  
       lv\_answer                   TYPE c,  
       lw\_usrgstin                 TYPE ty\_api,  
       lt\_eway\_upd\_transporter\_api TYPE ztt\_eway\_bill\_updat\_trn\_id\_str,  
       ls\_eway\_upd\_transporter\_api TYPE zst\_eway\_bill\_updat\_trn\_id\_str,  
       lt\_eway\_ext\_validity        TYPE ztt\_eway\_bill\_extend\_validity,  
       ls\_eway\_ext\_validity        TYPE zst\_eway\_bill\_extend\_validity,  
       lv\_return                   TYPE xfeld.  
  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  
  
  CLEAR: lt\_messages,lt\_ewaybill,lt\_eway\_upd\_transporter\_api.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
    PERFORM call\_popup USING   TEXT-t23 TEXT-t25  
                       CHANGING lv\_answer.  
  
    IF lv\_answer = '1'.  
      LOOP AT  gt\_index INTO lw\_index.  
  
        READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND gw\_final-eway\_num IS NOT INITIAL AND gw\_final-eway\_icon = gc\_icon\_08.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
          READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_eusrgstin.  
          IF sy-subrc IS INITIAL.  
            CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO ls\_eway\_upd\_transporter\_api-user\_gstin.  
          ELSE.  
            CONCATENATE '"' gw\_final-sup\_gstin '"' INTO ls\_eway\_upd\_transporter\_api-user\_gstin.  
          ENDIF.  
  
          ls\_eway\_upd\_transporter\_api-ebillno = gw\_final-eway\_num.  
  
          CONCATENATE '"' gw\_final-t\_id   '"' INTO ls\_eway\_upd\_transporter\_api-transporter\_id.  
          CONCATENATE '"' gw\_final-t\_name '"' INTO ls\_eway\_upd\_transporter\_api-transporter\_name.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
          APPEND ls\_eway\_upd\_transporter\_api TO lt\_eway\_upd\_transporter\_api.  
          CLEAR:ls\_eway\_upd\_transporter\_api.  
  
        ENDIF.  
      ENDLOOP.  
  
  
      IF lt\_eway\_upd\_transporter\_api IS NOT INITIAL.  
        SORT lt\_eway\_upd\_transporter\_api.  
  
        DELETE ADJACENT DUPLICATES FROM  lt\_eway\_upd\_transporter\_api  COMPARING ALL FIELDS.  
        CLEAR lv\_return.  
        CALL FUNCTION 'ZFM\_EWAY\_BILL\_VALIDATION'  
          EXPORTING  
            im\_ucomm         = gc\_ucomm\_ewayu  
            im\_api\_transport = lt\_eway\_upd\_transporter\_api  
          IMPORTING  
            ex\_error         = lv\_return.  
        IF lv\_return = ' '.  
  
          CLEAR:lw\_token,lw\_return.  
          IF gw\_token IS NOT INITIAL.  
            lw\_token = gw\_token.  
          ELSE.  
            CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
              IMPORTING  
                ex\_token    = lw\_token  
                ex\_return   = lw\_return  
                et\_messages = lt\_messages.  
          ENDIF.  
          IF lw\_token IS NOT INITIAL.  
            CALL FUNCTION 'ZFM\_EWAY\_BILL\_UPD\_TRANSID\_API'  
              EXPORTING  
                im\_token    = lw\_token  
                im\_api\_data = lt\_eway\_upd\_transporter\_api  
              IMPORTING  
                ex\_return   = lw\_return  
                ex\_messages = lt\_messages  
                ex\_api\_data = lt\_eway\_ext\_validity.  
  
            IF lt\_eway\_ext\_validity IS NOT INITIAL."lw\_return EQ 'S'.  
              LOOP AT lt\_eway\_ext\_validity INTO ls\_eway\_ext\_validity.  
                READ TABLE gt\_final INTO gw\_final WITH KEY eway\_num = ls\_eway\_ext\_validity-ewaybillno.  
                IF sy-subrc = 0.  
  
                  lw\_ewaybill-bukrs = gw\_final-bukrs.  
                  lw\_ewaybill-docno = gw\_final-vbeln.  
                  lw\_ewaybill-gjahr = gw\_final-gjahr.  
                  lw\_ewaybill-doctyp = gw\_final-fkart.  
                  lw\_ewaybill-ebillno = ls\_eway\_ext\_validity-ewaybillno.  
                  lw\_ewaybill-aenam     = sy-uname.  
                  lw\_ewaybill-aedat     = sy-datum.  
  
                  UPDATE j\_1ig\_ewaybill SET aenam    = lw\_ewaybill-aenam  
                                            aedat    = lw\_ewaybill-aedat  
                                          WHERE bukrs  = lw\_ewaybill-bukrs AND  
                                                docno  = lw\_ewaybill-docno AND  
                                                doctyp = lw\_ewaybill-doctyp AND  
                                                gjahr  = lw\_ewaybill-gjahr AND  
                                                ebillno = lw\_ewaybill-ebillno.  
                  IF sy-subrc = 0.  
                    APPEND  lw\_ewaybill TO lt\_ewaybill.  
                  ENDIF.  
                ENDIF.  
              ENDLOOP.  
  
              IF sy-subrc EQ 0.  
                COMMIT WORK.  
  
  
                CALL METHOD gref\_alv\_grid->refresh\_table\_display  
                  EXCEPTIONS  
                    finished = 1  
                    OTHERS   = 2.  
                IF sy-subrc <> 0.  
                  MESSAGE 'Table refresh error'(012) TYPE 'I'.  
                ENDIF.  
  
                CLEAR lt\_show\_message.  
                IF  lt\_messages IS NOT INITIAL.  
                  LOOP AT lt\_messages INTO lw\_message.  
                    lw\_show\_message-msgid =  gc\_msgid\_01.  
                    lw\_show\_message-msgty =  lw\_message-type.  
                    lw\_show\_message-msgno =  gc\_msgno\_319.  
                    lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
                    lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
                    lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
                    lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
                    APPEND lw\_show\_message TO lt\_show\_message.  
  
                  ENDLOOP.  
  
                  CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
                    TABLES  
                      i\_message\_tab = lt\_show\_message.  
  
                ENDIF.  
  
  
              ENDIF.  
  
            ENDIF.  
  
          ENDIF.  
        ENDIF.  
  
      ELSE.  
        MESSAGE 'Please select appropriate documents for E-Waybill update'(016) TYPE 'I'.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_EXTEND  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_extend .  
  
  
  DATA:lw\_index              TYPE lvc\_s\_row,  
       lv\_val                TYPE xfeld,  
       lv\_answer             TYPE c,  
       lw\_token              TYPE string,  
       lw\_return             TYPE string,  
       wa\_return             TYPE string,  
       lt\_messages           TYPE bapiret2\_t,  
       lw\_message            TYPE bapiret2,  
       lt\_show\_message       TYPE esp1\_message\_tab\_type,  
       lw\_show\_message       LIKE LINE OF lt\_show\_message,  
       lt\_ewaybill           TYPE TABLE OF j\_1ig\_ewaybill,  
       lw\_ewaybill           TYPE j\_1ig\_ewaybill,  
       lw\_wb2\_v\_vbrk\_vbrp2   TYPE ty\_wb2\_v\_vbrk\_vbrp2,  
       lt\_eway\_ext\_valid\_api TYPE ztt\_eway\_bill\_ext\_valid\_api,  
       ls\_eway\_ext\_valid\_api TYPE zst\_eway\_bill\_ext\_valid\_api,  
       lt\_eway\_ext\_validity  TYPE ztt\_eway\_bill\_extend\_validity,  
       ls\_eway\_ext\_validity  TYPE zst\_eway\_bill\_extend\_validity,  
       lw\_t001w              TYPE ty\_t001w1,  
       lw\_t005u\_s            TYPE ty\_t005u,  
       lw\_usrgstin           TYPE ty\_api,  
       ls\_values             TYPE dd07v,  
       lv\_return             TYPE xfeld.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  
  CLEAR: lt\_messages,lt\_ewaybill,lt\_eway\_ext\_valid\_api.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
    PERFORM call\_popup USING   TEXT-t23 TEXT-t26  
                       CHANGING lv\_answer.  
  
    IF lv\_answer = '1'.  
      LOOP AT  gt\_index INTO lw\_index.  
  
        READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND gw\_final-eway\_num IS NOT INITIAL.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
          ls\_eway\_ext\_valid\_api-ebillno = gw\_final-eway\_num.  
          READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_eusrgstin.  
          IF sy-subrc IS INITIAL.  
            CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO ls\_eway\_ext\_valid\_api-user\_gstin.  
          ELSE.  
            CONCATENATE '"' gw\_final-sup\_gstin '"' INTO ls\_eway\_ext\_valid\_api-user\_gstin.  
          ENDIF.  
  
          CONCATENATE '"' gw\_final-v\_number '"' INTO ls\_eway\_ext\_valid\_api-vehicle\_number.  
  
  
\*\*\* Supplier details  
          CLEAR: lw\_t001w.  
          READ TABLE gt\_t001w INTO lw\_t001w WITH  KEY werks = gw\_final-werks BINARY SEARCH.  
          IF sy-subrc EQ 0.  
  
            CONCATENATE '"' lw\_t001w-ort01 '"' INTO ls\_eway\_ext\_valid\_api-place\_of\_consignor.  
            CLEAR:lw\_t005u\_s.  
            READ TABLE gt\_t005u\_s INTO lw\_t005u\_s WITH  KEY bland = lw\_t001w-regio.  
            IF sy-subrc EQ 0.  
              TRANSLATE lw\_t005u\_s-bezei TO UPPER CASE.  
              CONCATENATE '"' lw\_t005u\_s-bezei  '"' INTO  ls\_eway\_ext\_valid\_api-state\_of\_consignor.  
            ENDIF.  
  
          ELSE.  
          ENDIF.  
  
          READ TABLE gt\_t\_mode\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-t\_mode.  
          IF sy-subrc EQ 0.  
            TRANSLATE ls\_values-ddtext TO UPPER CASE.  
            CONCATENATE '"' ls\_values-ddtext '"' INTO ls\_eway\_ext\_valid\_api-transportation\_mode.  
          ENDIF.  
          CONCATENATE '"' gw\_final-t\_r\_distance '"' INTO ls\_eway\_ext\_valid\_api-remaining\_distance.  
          CONCATENATE '"' gw\_final-t\_doc\_no '"' INTO ls\_eway\_ext\_valid\_api-transporter\_document\_number.  
          CONCATENATE '"' gw\_final-t\_date '"' INTO ls\_eway\_ext\_valid\_api-transporter\_document\_date.  
  
          READ TABLE gt\_t\_ext\_reason\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-t\_ext\_valid\_reason.  
          IF sy-subrc EQ 0.  
            TRANSLATE ls\_values-ddtext TO UPPER CASE.  
            CONCATENATE '"' ls\_values-ddtext '"' INTO ls\_eway\_ext\_valid\_api-extend\_validity\_reason.  
          ENDIF.  
  
          CONCATENATE '"' gw\_final-t\_ext\_valid\_remarks '"' INTO ls\_eway\_ext\_valid\_api-extend\_remarks.  
          ls\_eway\_ext\_valid\_api-from\_pincode  = gw\_final-t\_from\_pin.  
  
  
          CONCATENATE '"' gw\_final-t\_consignment\_status '"' INTO ls\_eway\_ext\_valid\_api-consignment\_status.  
  
  
\*& Transit is moving or not  
          IF gw\_final-eway\_v\_to GT sy-datum.  
  
            READ TABLE gt\_t\_transit\_type\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-t\_transit\_type.  
            IF sy-subrc EQ 0.  
              TRANSLATE ls\_values-ddtext TO UPPER CASE.  
              CONCATENATE '"' ls\_values-ddtext '"' INTO ls\_eway\_ext\_valid\_api-transit\_type.  
            ENDIF.  
            IF gw\_final-t\_transit\_type IS INITIAL.  
              CONCATENATE '"' ' ' '"' INTO ls\_eway\_ext\_valid\_api-transit\_type.  
            ENDIF.  
            CONCATENATE '"' gw\_final-t\_address1 '"' INTO ls\_eway\_ext\_valid\_api-address\_line1.  
            CONCATENATE '"' gw\_final-t\_address2 '"' INTO ls\_eway\_ext\_valid\_api-address\_line2.  
            CONCATENATE '"' gw\_final-t\_address3 '"' INTO ls\_eway\_ext\_valid\_api-address\_line3.  
          ELSE.  
            CONCATENATE '"' ' ' '"' INTO ls\_eway\_ext\_valid\_api-transit\_type.  
            CONCATENATE '"' ' ' '"' INTO ls\_eway\_ext\_valid\_api-address\_line1.  
            CONCATENATE '"' ' ' '"' INTO ls\_eway\_ext\_valid\_api-address\_line2.  
            CONCATENATE '"' ' ' '"' INTO ls\_eway\_ext\_valid\_api-address\_line3.  
          ENDIF.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
          APPEND ls\_eway\_ext\_valid\_api TO lt\_eway\_ext\_valid\_api.  
          CLEAR:ls\_eway\_ext\_valid\_api.  
  
        ENDIF.  
      ENDLOOP.  
  
  
      IF lt\_eway\_ext\_valid\_api IS NOT INITIAL.  
        SORT lt\_eway\_ext\_valid\_api.  
  
        DELETE ADJACENT DUPLICATES FROM  lt\_eway\_ext\_valid\_api  COMPARING ALL FIELDS.  
        CALL FUNCTION 'ZFM\_EWAY\_BILL\_VALIDATION'  
          EXPORTING  
            im\_ucomm      = gc\_ucomm\_ewaye  
            im\_api\_extend = lt\_eway\_ext\_valid\_api  
          IMPORTING  
            ex\_error      = lv\_return.  
  
        CLEAR:lw\_token,lw\_return.  
        IF gw\_token IS NOT INITIAL.  
          lw\_token = gw\_token.  
        ELSE.  
          CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
            IMPORTING  
              ex\_token    = lw\_token  
              ex\_return   = lw\_return  
              et\_messages = lt\_messages.  
        ENDIF.  
        IF lw\_token IS NOT INITIAL." AND lw\_return EQ 'S'.  
  
          CALL FUNCTION 'ZFM\_EWAY\_BILL\_EXTEND\_VALID\_API'  
            EXPORTING  
              im\_token       = lw\_token  
              im\_api\_data    = lt\_eway\_ext\_valid\_api  
            IMPORTING  
              ex\_return      = lw\_return  
              ex\_messages    = lt\_messages  
              ex\_exd\_details = lt\_eway\_ext\_validity.  
  
  
          IF lt\_eway\_ext\_validity IS NOT INITIAL."lw\_return EQ 'S'.  
            LOOP AT lt\_eway\_ext\_validity INTO ls\_eway\_ext\_validity.  
              READ TABLE gt\_final ASSIGNING <gw\_final> WITH KEY eway\_num = ls\_eway\_ext\_validity-ewaybillno.  
              IF sy-subrc = 0.  
                IF ls\_eway\_ext\_validity-validupto IS NOT INITIAL.  
                  lw\_ewaybill-bukrs = <gw\_final>-bukrs.  
                  lw\_ewaybill-docno = <gw\_final>-vbeln.  
                  lw\_ewaybill-gjahr = <gw\_final>-gjahr.  
                  lw\_ewaybill-doctyp = <gw\_final>-fkart.  
                  lw\_ewaybill-ebillno = ls\_eway\_ext\_validity-ewaybillno.  
                  lw\_ewaybill-vdtodate =   ls\_eway\_ext\_validity-validupto.  
                  lw\_ewaybill-aenam     = sy-uname.  
                  lw\_ewaybill-aedat     = sy-datum.  
  
                  UPDATE j\_1ig\_ewaybill SET vdtodate = lw\_ewaybill-vdtodate  
                                            aenam    = lw\_ewaybill-aenam  
                                            aedat    = lw\_ewaybill-aedat  
                                          WHERE bukrs  = lw\_ewaybill-bukrs AND  
                                                docno  = lw\_ewaybill-docno AND  
                                                doctyp = lw\_ewaybill-doctyp AND  
                                                gjahr  = lw\_ewaybill-gjahr AND  
                                                ebillno = lw\_ewaybill-ebillno.  
                  IF sy-subrc = 0.  
                    CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                      EXPORTING  
                        date\_internal            = lw\_ewaybill-vdtodate  
                      IMPORTING  
                        date\_external            = <gw\_final>-eway\_v\_to  
                      EXCEPTIONS  
                        date\_internal\_is\_invalid = 1  
                        OTHERS                   = 2.  
                    IF sy-subrc <> 0.  
                      MESSAGE 'Data Conversion is incomplete' TYPE 'I'.  
                    ENDIF.  
                  ENDIF.  
                ENDIF.  
              ENDIF.  
            ENDLOOP.  
  
            IF sy-subrc EQ 0.  
              COMMIT WORK.  
  
  
              CALL METHOD gref\_alv\_grid->refresh\_table\_display  
                EXCEPTIONS  
                  finished = 1  
                  OTHERS   = 2.  
              IF sy-subrc <> 0.  
                MESSAGE 'Table refresh error'(012) TYPE 'I'.  
              ENDIF.  
  
              CLEAR lt\_show\_message.  
              IF  lt\_messages IS NOT INITIAL.  
                LOOP AT lt\_messages INTO lw\_message.  
                  lw\_show\_message-msgid =  gc\_msgid\_01.  
                  lw\_show\_message-msgty =  lw\_message-type.  
                  lw\_show\_message-msgno =  gc\_msgno\_319.  
                  lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
                  lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
                  lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
                  lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
                  APPEND lw\_show\_message TO lt\_show\_message.  
  
                ENDLOOP.  
  
                CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
                  TABLES  
                    i\_message\_tab = lt\_show\_message.  
  
              ENDIF.  
  
  
            ENDIF.  
  
          ENDIF.  
  
        ENDIF.  
  
  
      ELSE.  
        MESSAGE 'Please select appropriate documents for E-Waybill extension'(017) TYPE 'I'.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_CANCEL  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_cancel USING p\_lv\_reason\_code  
                           p\_lv\_reason.  
  
  DATA:lw\_token           TYPE string,  
       lw\_return          TYPE string,  
       wa\_return          TYPE string,  
       lt\_messages        TYPE bapiret2\_t,  
       lw\_message         TYPE bapiret2,  
       lt\_show\_message    TYPE esp1\_message\_tab\_type,  
       lw\_show\_message    LIKE LINE OF lt\_show\_message,  
       lt\_ewaybill        TYPE TABLE OF j\_1ig\_ewaybill,  
       lw\_ewaybill        TYPE j\_1ig\_ewaybill,  
       lt\_eway\_cancel\_api TYPE ztt\_eway\_bill\_cancel\_str,  
       ls\_eway\_cancel\_api TYPE zst\_eway\_bill\_cancel\_str,  
       lt\_eway\_cancel     TYPE ztt\_eway\_bill\_cancel\_out\_str,  
       ls\_eway\_cancel     TYPE zst\_eway\_bill\_cancel\_out\_str,  
       lw\_usrgstin        TYPE ty\_api,  
       ls\_values          TYPE dd07v.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  
  CLEAR: lt\_messages,lt\_ewaybill,lt\_eway\_cancel\_api.  
  
  
  LOOP AT  gt\_index INTO lw\_index.  
  
    READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
    IF sy-subrc IS INITIAL AND gw\_final-eway\_num IS NOT INITIAL.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
      READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_eusrgstin.  
      IF sy-subrc IS INITIAL.  
        CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO ls\_eway\_cancel\_api-user\_gstin.  
      ELSE.  
        CONCATENATE '"' gw\_final-sup\_gstin '"' INTO ls\_eway\_cancel\_api-user\_gstin.  
      ENDIF.  
  
      ls\_eway\_cancel\_api-ebillno = gw\_final-eway\_num.  
  
      ls\_eway\_cancel\_api-data\_source = gc\_dsource\_erp.  
  
      READ TABLE gt\_cancel\_values INTO ls\_values WITH KEY domvalue\_l = p\_lv\_reason\_code.  
      IF sy-subrc EQ 0.  
        TRANSLATE ls\_values-ddtext TO UPPER CASE.  
        CONCATENATE '"' ls\_values-ddtext   '"' INTO ls\_eway\_cancel\_api-reason\_of\_cancel.  
      ENDIF.  
      CONCATENATE '"' p\_lv\_reason       '"' INTO ls\_eway\_cancel\_api-cancel\_remark.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
      APPEND ls\_eway\_cancel\_api TO lt\_eway\_cancel\_api.  
      CLEAR:ls\_eway\_cancel\_api.  
  
    ENDIF.  
  ENDLOOP.  
  
  
  IF lt\_eway\_cancel\_api IS NOT INITIAL.  
    SORT lt\_eway\_cancel\_api.  
  
    DELETE ADJACENT DUPLICATES FROM  lt\_eway\_cancel\_api  COMPARING ALL FIELDS.  
  
    CLEAR:lw\_token,lw\_return.  
    IF gw\_token IS NOT INITIAL.  
      lw\_token = gw\_token.  
    ELSE.  
      CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
        IMPORTING  
          ex\_token    = lw\_token  
          ex\_return   = lw\_return  
          et\_messages = lt\_messages.  
    ENDIF.  
    IF lw\_token IS NOT INITIAL.  
  
      CALL FUNCTION 'ZFM\_EWAY\_BILL\_CANCEL\_API'  
        EXPORTING  
          im\_token           = lw\_token  
          im\_api\_data        = lt\_eway\_cancel\_api  
        IMPORTING  
          ex\_return          = lw\_return  
          ex\_messages        = lt\_messages  
          ex\_ewaybill\_cancel = lt\_eway\_cancel.  
  
  
      IF lt\_eway\_cancel IS NOT INITIAL.  
        LOOP AT lt\_eway\_cancel INTO ls\_eway\_cancel.  
          READ TABLE gt\_final ASSIGNING  <gw\_final> WITH KEY eway\_num = ls\_eway\_cancel-ewaybillno.  
          IF sy-subrc = 0.  
  
            lw\_ewaybill-bukrs = gw\_final-bukrs.  
            lw\_ewaybill-docno = gw\_final-vbeln.  
            lw\_ewaybill-gjahr = gw\_final-gjahr.  
            lw\_ewaybill-doctyp = gw\_final-fkart.  
            lw\_ewaybill-ebillno = ls\_eway\_cancel-ewaybillno.  
            lw\_ewaybill-aenam     = sy-uname.  
            lw\_ewaybill-aedat     = sy-datum.  
            lw\_ewaybill-status    = gw\_final-eway\_status.  
  
            IF ls\_eway\_cancel-canceldate IS NOT INITIAL.  
              lw\_ewaybill-status     = gc\_eway\_sts\_c.  
              UPDATE j\_1ig\_ewaybill SET status   = lw\_ewaybill-status  
                                        aenam    = lw\_ewaybill-aenam  
                                        aedat    = lw\_ewaybill-aedat  
                                      WHERE bukrs  = lw\_ewaybill-bukrs AND  
                                            docno  = lw\_ewaybill-docno AND  
                                            doctyp = lw\_ewaybill-doctyp AND  
                                            gjahr  = lw\_ewaybill-gjahr AND  
                                            ebillno = lw\_ewaybill-ebillno.  
              IF sy-subrc = 0.  
                UPDATE zteway\_transport SET c\_reason\_code = p\_lv\_reason\_code  
                                            c\_reason      = p\_lv\_reason  
                                            aenam         = lw\_ewaybill-aenam  
                                            aedat         = lw\_ewaybill-aedat  
                                          WHERE  bukrs  = lw\_ewaybill-bukrs AND  
                                            docno  = lw\_ewaybill-docno AND  
                                            doctyp = lw\_ewaybill-doctyp AND  
                                            gjahr  = lw\_ewaybill-gjahr.  
  
  
                IF sy-subrc IS INITIAL.  
                  <gw\_final>-eway\_status = 'Cancelled'(062).  
                  <gw\_final>-eway\_icon    = gc\_icon\_0w.  
  
                  CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                    EXPORTING  
                      date\_internal            = sy-datum  
                    IMPORTING  
                      date\_external            = <gw\_final>-eway\_canc\_dt  
                    EXCEPTIONS  
                      date\_internal\_is\_invalid = 1  
                      OTHERS                   = 2.  
                  IF sy-subrc <> 0.  
                    MESSAGE 'Data Conversion is incomplete' TYPE 'I'.  
                  ENDIF.  
                ENDIF.  
              ENDIF.  
            ENDIF.  
          ENDIF.  
        ENDLOOP.  
  
        COMMIT WORK.  
  
  
        CALL METHOD gref\_alv\_grid->refresh\_table\_display  
          EXCEPTIONS  
            finished = 1  
            OTHERS   = 2.  
        IF sy-subrc <> 0.  
          MESSAGE 'Table refresh error'(012) TYPE 'I'.  
        ENDIF.  
  
        CLEAR lt\_show\_message.  
        IF  lt\_messages IS NOT INITIAL.  
          LOOP AT lt\_messages INTO lw\_message.  
            lw\_show\_message-msgid =  gc\_msgid\_01.  
            lw\_show\_message-msgty =  lw\_message-type.  
            lw\_show\_message-msgno =  gc\_msgno\_319.  
            lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
            lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
            lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
            lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
            APPEND lw\_show\_message TO lt\_show\_message.  
  
          ENDLOOP.  
  
          CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
            TABLES  
              i\_message\_tab = lt\_show\_message.  
  
        ENDIF.  
  
      ENDIF.  
  
    ENDIF.  
  
  
  ELSE.  
    MESSAGE 'Please select appropriate documents to cancel E-Waybill'(018) TYPE 'I'.  
  ENDIF.  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_UPDATE\_VEHICLE  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_update\_vehicle .  
  
  
  DATA:lw\_index  TYPE lvc\_s\_row,  
       lv\_val    TYPE xfeld,  
       lv\_answer TYPE c.  
  
  DATA:lv\_index        TYPE sy-tabix.  
  
  DATA:lw\_token        TYPE string,  
       lw\_return       TYPE string,  
       wa\_return       TYPE string,  
       lt\_messages     TYPE bapiret2\_t,  
       lw\_message      TYPE bapiret2,  
       lt\_show\_message TYPE esp1\_message\_tab\_type,  
       lw\_show\_message LIKE LINE OF lt\_show\_message.  
  
  DATA:lt\_ewaybill TYPE TABLE OF j\_1ig\_ewaybill,  
       lw\_ewaybill TYPE j\_1ig\_ewaybill.  
  
  DATA :lw\_wb2\_v\_vbrk\_vbrp2 TYPE ty\_wb2\_v\_vbrk\_vbrp2.  
  
  DATA: lt\_eway\_upd\_vehile\_api TYPE ztt\_eway\_bill\_updat\_vehno\_str,  
        ls\_eway\_upd\_vehile\_api TYPE zst\_eway\_bill\_updat\_vehno\_str.  
  
  DATA: lt\_eway\_ext\_validity TYPE ztt\_eway\_bill\_extend\_validity,  
        ls\_eway\_ext\_validity TYPE zst\_eway\_bill\_extend\_validity.  
  
  
  DATA: lw\_t001w   TYPE ty\_t001w1,  
        lw\_t005u\_s TYPE ty\_t005u.  
  
  DATA:lw\_usrgstin TYPE ty\_api,  
       lv\_datum    TYPE sydatum.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  DATA: ls\_values  TYPE dd07v.  
  DATA:lv\_return TYPE xfeld.  
  CLEAR: lt\_messages,lt\_ewaybill,lt\_eway\_upd\_vehile\_api.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
    PERFORM call\_popup USING   TEXT-t23 TEXT-t25  
                       CHANGING lv\_answer.  
  
    IF lv\_answer = '1'.  
      LOOP AT  gt\_index INTO lw\_index.  
  
        READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND gw\_final-eway\_num IS NOT INITIAL AND gw\_final-eway\_icon = gc\_icon\_08..  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
          READ TABLE gt\_api  INTO lw\_usrgstin WITH KEY apiid = gc\_apiid\_eusrgstin.  
          IF sy-subrc IS INITIAL.  
            CONCATENATE '"' lw\_usrgstin-apiprov '"' INTO ls\_eway\_upd\_vehile\_api-user\_gstin.  
          ELSE.  
            CONCATENATE '"' gw\_final-sup\_gstin '"' INTO ls\_eway\_upd\_vehile\_api-user\_gstin.  
          ENDIF.  
  
          ls\_eway\_upd\_vehile\_api-ebillno = gw\_final-eway\_num.  
  
          ls\_eway\_upd\_vehile\_api-data\_source = gc\_dsource\_erp.  
          CONCATENATE '"' gw\_final-v\_number '"' INTO ls\_eway\_upd\_vehile\_api-vehicle\_number.  
          CLEAR ls\_values.  
          READ TABLE gt\_v\_type\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-v\_type.  
          IF sy-subrc EQ 0.  
            TRANSLATE ls\_values-ddtext TO UPPER CASE.  
            CONCATENATE '"' ls\_values-ddtext   '"' INTO ls\_eway\_upd\_vehile\_api-vehicle\_type.  
          ENDIF.  
\*\*\* Supplier details  
          CLEAR: lw\_t001w. ", lw\_adrc\_s, lw\_adr6\_s.  
          READ TABLE gt\_t001w INTO lw\_t001w WITH  KEY werks = gw\_final-werks BINARY SEARCH.  
          IF sy-subrc EQ 0.  
  
            CONCATENATE '"' lw\_t001w-ort01 '"' INTO ls\_eway\_upd\_vehile\_api-place\_of\_consignor.  
  
            CLEAR:lw\_t005u\_s.  
            READ TABLE gt\_t005u\_s INTO lw\_t005u\_s WITH  KEY bland = lw\_t001w-regio.  
            IF sy-subrc EQ 0.  
              TRANSLATE lw\_t005u\_s-bezei TO UPPER CASE.  
              CONCATENATE '"' lw\_t005u\_s-bezei  '"' INTO  ls\_eway\_upd\_vehile\_api-state\_of\_consignor.  
            ENDIF.  
          ENDIF.  
  
\*\*\* Transport details  
          READ TABLE gt\_t\_mode\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-t\_mode.  
          IF sy-subrc EQ 0.  
            TRANSLATE ls\_values-ddtext TO UPPER CASE.  
            CONCATENATE '"' ls\_values-ddtext '"' INTO ls\_eway\_upd\_vehile\_api-transportation\_mode.  
          ENDIF.  
          CONCATENATE '"' gw\_final-t\_doc\_no '"' INTO ls\_eway\_upd\_vehile\_api-transporter\_document\_number.  
          CONCATENATE '"' gw\_final-t\_date '"' INTO ls\_eway\_upd\_vehile\_api-transporter\_document\_date.  
  
          READ TABLE gt\_v\_reason\_values INTO ls\_values WITH KEY domvalue\_l = gw\_final-v\_reason\_code.  
          IF sy-subrc EQ 0.  
            TRANSLATE ls\_values-ddtext TO UPPER CASE.  
            CONCATENATE '"' ls\_values-ddtext '"' INTO ls\_eway\_upd\_vehile\_api-reason\_code\_for\_vehicle\_updati.  
          ENDIF.  
  
          CONCATENATE '"' gw\_final-v\_reason '"' INTO ls\_eway\_upd\_vehile\_api-reason\_for\_vehicle\_updation.  
  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
          APPEND ls\_eway\_upd\_vehile\_api TO lt\_eway\_upd\_vehile\_api.  
          CLEAR:ls\_eway\_upd\_vehile\_api.  
  
        ENDIF.  
      ENDLOOP.  
  
  
      IF lt\_eway\_upd\_vehile\_api IS NOT INITIAL.  
        SORT lt\_eway\_upd\_vehile\_api.  
  
        DELETE ADJACENT DUPLICATES FROM  lt\_eway\_upd\_vehile\_api  COMPARING ALL FIELDS.  
        CLEAR lv\_return.  
        CALL FUNCTION 'ZFM\_EWAY\_BILL\_VALIDATION'  
          EXPORTING  
            im\_ucomm       = gc\_ucomm\_ewayv  
            im\_api\_vehicle = lt\_eway\_upd\_vehile\_api  
          IMPORTING  
            ex\_error       = lv\_return.  
        IF lv\_return = ' '.  
          CLEAR:lw\_token,lw\_return.  
          IF gw\_token IS NOT INITIAL.  
            lw\_token = gw\_token.  
          ELSE.  
            CALL FUNCTION 'ZFM\_EINVOICE\_OAUTH\_API'  
              IMPORTING  
                ex\_token    = lw\_token  
                ex\_return   = lw\_return  
                et\_messages = lt\_messages.  
          ENDIF.  
          IF lw\_token IS NOT INITIAL." AND lw\_return EQ 'S'.  
  
            CALL FUNCTION 'ZFM\_EWAY\_BILL\_UPD\_VEHL\_NO\_API'  
              EXPORTING  
                im\_token    = lw\_token  
                im\_api\_data = lt\_eway\_upd\_vehile\_api  
              IMPORTING  
                ex\_return   = lw\_return  
                ex\_messages = lt\_messages  
                ex\_api\_data = lt\_eway\_ext\_validity.  
  
  
            IF lt\_eway\_ext\_validity IS NOT INITIAL."lw\_return EQ 'S'.  
              LOOP AT lt\_eway\_ext\_validity INTO ls\_eway\_ext\_validity.  
                READ TABLE gt\_final ASSIGNING <gw\_final> WITH KEY eway\_num = ls\_eway\_ext\_validity-ewaybillno.  
                IF sy-subrc = 0.  
  
                  IF ls\_eway\_ext\_validity-validupto IS NOT INITIAL.  
  
  
                    lw\_ewaybill-bukrs = gw\_final-bukrs.  
                    lw\_ewaybill-docno = gw\_final-vbeln.  
                    lw\_ewaybill-gjahr = gw\_final-gjahr.  
                    lw\_ewaybill-doctyp = gw\_final-fkart.  
                    lw\_ewaybill-ebillno = ls\_eway\_ext\_validity-ewaybillno.  
                    lw\_ewaybill-vdtodate = ls\_eway\_ext\_validity-validupto.  
                    lw\_ewaybill-vdtotime = ls\_eway\_ext\_validity-vdtotime.  
  
  
                    UPDATE j\_1ig\_ewaybill SET vdtodate  = lw\_ewaybill-vdtodate  
                                              vdtotime  = lw\_ewaybill-vdtotime  
                                              aenam    = sy-uname  
                                              aedat    = sy-datum  
                                            WHERE bukrs  = lw\_ewaybill-bukrs AND  
                                                  docno  = lw\_ewaybill-docno AND  
                                                  doctyp = lw\_ewaybill-doctyp AND  
                                                  gjahr  = lw\_ewaybill-gjahr AND  
                                                  ebillno = lw\_ewaybill-ebillno.  
                    IF sy-subrc = 0.  
  
                      CALL FUNCTION 'CONVERT\_DATE\_TO\_EXTERNAL'  
                        EXPORTING  
                          date\_internal            = lw\_ewaybill-vdtodate  
                        IMPORTING  
                          date\_external            = <gw\_final>-eway\_v\_to  
                        EXCEPTIONS  
                          date\_internal\_is\_invalid = 1  
                          OTHERS                   = 2.  
                      IF sy-subrc <> 0.  
                        MESSAGE 'Date convertion error'(010) TYPE 'I'.  
                      ENDIF.  
  
  
                    ENDIF.  
                  ENDIF.  
                ENDIF.  
              ENDLOOP.  
  
              IF sy-subrc EQ 0.  
                COMMIT WORK.  
  
                CALL METHOD gref\_alv\_grid->refresh\_table\_display  
                  EXCEPTIONS  
                    finished = 1  
                    OTHERS   = 2.  
                IF sy-subrc <> 0.  
                  MESSAGE 'Table refresh error'(012) TYPE 'I'.  
                ENDIF.  
  
                CLEAR lt\_show\_message.  
                IF  lt\_messages IS NOT INITIAL.  
                  LOOP AT lt\_messages INTO lw\_message.  
                    lw\_show\_message-msgid =  gc\_msgid\_01.  
                    lw\_show\_message-msgty =  lw\_message-type.  
                    lw\_show\_message-msgno =  gc\_msgno\_319.  
                    lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
                    lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
                    lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
                    lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
                    APPEND lw\_show\_message TO lt\_show\_message.  
  
                  ENDLOOP.  
  
                  CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
                    TABLES  
                      i\_message\_tab = lt\_show\_message.  
  
                ENDIF.  
  
  
              ENDIF.  
  
            ENDIF.  
  
          ENDIF.  
  
        ENDIF.  
      ELSE.  
        MESSAGE 'Please select appropriate documents for E-Waybill update'(016) TYPE 'I'.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_PRINT  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_print .  
  
  DATA:lw\_index  TYPE lvc\_s\_row,  
       lv\_val    TYPE xfeld,  
       lv\_answer TYPE c.  
  
  DATA:lv\_index        TYPE sy-tabix.  
  
  DATA:lw\_token        TYPE string,  
       lw\_return       TYPE string,  
       wa\_return       TYPE string,  
       lt\_messages     TYPE bapiret2\_t,  
       lw\_message      TYPE bapiret2,  
       lt\_show\_message TYPE esp1\_message\_tab\_type,  
       lw\_show\_message LIKE LINE OF lt\_show\_message.  
  
  DATA: lv\_url TYPE string.  
  
  DATA: lo\_http\_client TYPE REF TO if\_http\_client,  
        lv\_service     TYPE string,  
        lv\_result      TYPE string,  
        lo\_ixml        TYPE REF TO if\_ixml.  
  
  
\*  DATA: ls\_config TYPE zteinv\_api.  
  
  DATA:lt\_ewaybill TYPE TABLE OF j\_1ig\_ewaybill,  
       lw\_ewaybill TYPE j\_1ig\_ewaybill.  
  
  DATA:lw\_eway\_transport TYPE zteway\_transport.  
  
  FIELD-SYMBOLS <gw\_final>  TYPE ty\_final.  
  
  
  CLEAR: lt\_messages,lt\_ewaybill.  
  
  
  CLEAR lv\_url.  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
    PERFORM call\_popup USING   TEXT-t23 TEXT-t29  
                       CHANGING lv\_answer.  
  
    IF lv\_answer = '1'.  
  
      LOOP AT  gt\_index INTO lw\_index.  
  
        READ TABLE gt\_final INTO gw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND gw\_final-eway\_num IS NOT INITIAL.  
  
          CONCATENATE 'https://'(109) gw\_final-eway\_print '.pdf'(110) INTO lv\_url.  
          CONDENSE lv\_url NO-GAPS.  
  
          CALL METHOD cl\_gui\_frontend\_services=>execute  
            EXPORTING  
              document = lv\_url  
            EXCEPTIONS  
              OTHERS   = 1.  
          IF sy-subrc <> 0.  
            MESSAGE 'Pdf couldn not find'(023) TYPE 'I'.  
          ENDIF.  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Begin of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*   End of Fill header derails   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
        ENDIF.  
      ENDLOOP.  
  
  
      CLEAR lt\_show\_message.  
      IF  lt\_messages IS NOT INITIAL.  
        LOOP AT lt\_messages INTO lw\_message.  
          lw\_show\_message-msgid =  '01'.  
          lw\_show\_message-msgty =  lw\_message-type.  
          lw\_show\_message-msgno =  '319'.  
          lw\_show\_message-msgv1 =  lw\_message-message\_v1.  
          lw\_show\_message-msgv2 =  lw\_message-message\_v2.  
          lw\_show\_message-msgv3 =  lw\_message-message\_v3.  
          lw\_show\_message-msgv4 =  lw\_message-message\_v4.  
          APPEND lw\_show\_message TO lt\_show\_message.  
  
        ENDLOOP.  
  
        CALL FUNCTION 'C14Z\_MESSAGES\_SHOW\_AS\_POPUP'  
          TABLES  
            i\_message\_tab = lt\_show\_message.  
  
      ENDIF.  
  
    ENDIF.  
  
  
  ELSE.  
    MESSAGE 'Please select appropriate documents for E-Waybill print'(020) TYPE 'I'.  
  ENDIF.  
\*  ENDIF.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  USER\_COMMAND\_EWAY\_CANCEL  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM user\_command\_eway\_cancel TABLES ifields TYPE ty\_sval  
                              USING  ok\_code\_save  
                                     error  
                                     h\_show\_popup.  
  DATA: ls\_ifields TYPE sval.  
  DATA:lv\_reason\_code TYPE zde\_cancel\_reason,  
       lv\_reason      TYPE char120.  
  
  CASE ok\_code\_save.  
    WHEN gc\_okcode\_furt.  
      LOOP AT ifields INTO  ls\_ifields.  
        IF ls\_ifields-fieldname = gc\_fname\_crec\_code."'C\_REASON\_CODE'.  
          lv\_reason\_code = ls\_ifields-value.  
        ELSEIF ls\_ifields-fieldname = gc\_fname\_crec."'C\_REASON'.  
          lv\_reason = ls\_ifields-value.  
        ENDIF.  
      ENDLOOP.  
      PERFORM ewaybill\_cancel USING lv\_reason\_code lv\_reason.  
    WHEN OTHERS.  
  ENDCASE.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_CANCEL\_OLD  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM ewaybill\_cancel\_old .  
  
  
  DATA:        lv\_val    TYPE xfeld.  
  DATA: vl\_program      TYPE sy-repid,  
        vl\_formname     TYPE char30,  
        vl\_popup\_title  TYPE char30,  
        vl\_okpushbutton TYPE svalbutton-buttontext,  
        vl\_buttonicon   TYPE icon-name,  
        lt\_sval         TYPE ty\_sval,  
        lv\_msg          TYPE xfeld,  
        lv\_answer       TYPE c,  
        lw\_final        TYPE ty\_final.  
  
  FIELD-SYMBOLS:<ls\_sval> TYPE sval.  
  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
  
    PERFORM call\_popup USING   TEXT-t23 TEXT-t22  
                      CHANGING lv\_answer.  
  
    IF lv\_answer = '1'.  
      CLEAR lv\_msg.  
      LOOP AT gt\_index INTO lw\_index.  
        READ TABLE gt\_final INTO lw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND lw\_final-eway\_icon = gc\_icon\_0w.  
          MESSAGE 'Document already cancelled'(037) TYPE 'I'.  
          lv\_msg = abap\_true.  
          EXIT.  
        ENDIF.  
      ENDLOOP.  
      IF lv\_msg IS INITIAL.  
\*& Add title & FORMNAME for Cancel pop-up  
        vl\_program        = sy-repid.  
        vl\_formname       = 'USER\_COMMAND\_EWAY\_CANCEL'.  
        vl\_popup\_title    = 'Confirm to cancel E-way Bill'(114).  
        vl\_okpushbutton   = 'Cancel E-Way Bill'(116).  
        vl\_buttonicon     = gc\_icon\_b2.  
\*& Add two fields to input cancel reason  
        APPEND INITIAL LINE TO lt\_sval ASSIGNING <ls\_sval>.  
        <ls\_sval>-tabname = 'ZTEWAY\_TRANSPORT'.  
        <ls\_sval>-fieldname = 'C\_REASON\_CODE'.  
        <ls\_sval>-field\_obl = abap\_true.  
        <ls\_sval>-fieldtext = 'Reason for cancel'(113).  
        APPEND INITIAL LINE TO lt\_sval ASSIGNING <ls\_sval>.  
        <ls\_sval>-tabname = 'ZTEWAY\_TRANSPORT'.  
        <ls\_sval>-fieldname = 'C\_REASON'.  
        <ls\_sval>-field\_obl = abap\_true.  
        <ls\_sval>-fieldtext = 'Remarks'(115).  
        PERFORM popup\_to\_confirm\_with\_input USING vl\_program  
                                                  vl\_formname  
                                                  vl\_popup\_title  
                                                  vl\_okpushbutton  
                                                  vl\_buttonicon  
                                                  lt\_sval.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*&      Form  EWAYBILL\_CANCEL\_OLD  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM einvoice\_cancel\_old .  
  
  
  DATA:lv\_val          TYPE xfeld,  
       vl\_program      TYPE sy-repid,  
       vl\_formname     TYPE char30,  
       vl\_popup\_title  TYPE char30,  
       vl\_okpushbutton TYPE svalbutton-buttontext,  
       vl\_buttonicon   TYPE icon-name,  
       lt\_sval         TYPE ty\_sval,  
       lw\_index        TYPE lvc\_s\_row,  
       lv\_msg          TYPE xfeld,  
       lv\_answer       TYPE c,  
       lw\_final        TYPE ty\_final.  
  
  
  FIELD-SYMBOLS:<ls\_sval> TYPE sval.  
  
  
  PERFORM check\_selected\_data CHANGING lv\_val.  
  
  IF lv\_val IS INITIAL.  
  
    PERFORM call\_popup USING   TEXT-t20 TEXT-t22  
                       CHANGING lv\_answer.  
  
    IF lv\_answer = 1.  
      CLEAR lv\_msg.  
      LOOP AT gt\_index INTO lw\_index.  
        READ TABLE gt\_final INTO lw\_final INDEX lw\_index-index.  
        IF sy-subrc IS INITIAL AND lw\_final-icon = gc\_icon\_0w.  
          MESSAGE 'Document already cancelled'(037) TYPE 'I'.  
          lv\_msg = abap\_true.  
          EXIT.  
        ENDIF.  
      ENDLOOP.  
  
      IF lv\_msg IS INITIAL.  
  
\*& Add title & FORMNAME for Cancel pop-up  
        vl\_program        = sy-repid.  
        vl\_formname       = 'USER\_COMMAND\_EINV\_CANCEL'.  
        vl\_popup\_title    = 'Confirm to cancel E-Invoice'(111).  
        vl\_okpushbutton   = 'Cancel E-Invoice'(112).  
        vl\_buttonicon     = '@B2@'."gc\_icon\_b2.  
\*& Add two fields to input cancel reason  
        APPEND INITIAL LINE TO lt\_sval ASSIGNING <ls\_sval>.  
        <ls\_sval>-tabname = 'ZTEINV\_DETAILS'."gc\_tname\_einv.  
        <ls\_sval>-fieldname = 'E\_REASON\_CODE'.  
        <ls\_sval>-field\_obl = abap\_true.  
        <ls\_sval>-fieldtext = 'Reason for cancel'(113).  
        APPEND INITIAL LINE TO lt\_sval ASSIGNING <ls\_sval>.  
        <ls\_sval>-tabname = 'ZTEINV\_DETAILS'.  
        <ls\_sval>-fieldname = 'E\_REASON'.  
        <ls\_sval>-field\_obl = abap\_true.  
        <ls\_sval>-fieldtext = 'Remarks'(115).  
        PERFORM popup\_to\_confirm\_with\_input USING vl\_program  
                                                  vl\_formname  
                                                  vl\_popup\_title  
                                                  vl\_okpushbutton  
                                                  vl\_buttonicon  
                                                  lt\_sval.  
      ENDIF.  
    ENDIF.  
  ENDIF.  
ENDFORM.  
\*&      Form  USER\_COMMAND\_EINVOICE\_CANCEL  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM user\_command\_einv\_cancel TABLES ifields TYPE ty\_sval  
                              USING  ok\_code\_save  
                                     error  
                                     h\_show\_popup.  
  DATA: ls\_ifields TYPE sval.  
  DATA:lv\_reason\_code TYPE zde\_cancel\_reason,  
       lv\_reason      TYPE char120.  
  
  CASE ok\_code\_save.  
    WHEN gc\_okcode\_furt.  
      LOOP AT ifields INTO  ls\_ifields.  
        IF ls\_ifields-fieldname = 'E\_REASON\_CODE'.  
          lv\_reason\_code = ls\_ifields-value.  
        ELSEIF ls\_ifields-fieldname =  'E\_REASON'.  
          lv\_reason = ls\_ifields-value.  
        ENDIF.  
      ENDLOOP.  
      PERFORM einvoice\_cancel USING lv\_reason\_code lv\_reason.  
    WHEN OTHERS.  
  ENDCASE.  
  
ENDFORM.