Program#1: All Fields using Database Table Name

\*&---------------------------------------------------------------------\*  
\*& Report  ZEXAMPLE\_ALV\_01  
\*&  
\*&---------------------------------------------------------------------\*  
\*&  
\*&  
\*&---------------------------------------------------------------------\*  
  
REPORT zexample\_alv\_01.  
DATA:  
   lv\_vbeln TYPE vbeln,  
   lv\_kunnr TYPE kunnr.  
PARAMETERS: p\_vkorg TYPE vkorg.  
SELECT-OPTIONS:  
s\_vbeln  FOR lv\_vbeln,  
s\_kunnr FOR lv\_kunnr.  
  
DATA: it\_vbak TYPE TABLE OF vbak.  
  
START-OF-SELECTION.  
  IF p\_vkorg IS NOT INITIAL  
  OR s\_vbeln IS NOT INITIAL  
  OR s\_kunnr IS NOT INITIAL.  
    SELECT \*  
      FROM vbak  
      INTO TABLE it\_vbak  
      WHERE vbeln IN s\_vbeln  
      AND vkorg = p\_vkorg  
      AND kunnr IN s\_kunnr.  
    IF it\_vbak IS NOT INITIAL.  
      CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
        EXPORTING  
          i\_callback\_program = sy-repid  
          i\_structure\_name   = 'VBAK'  
        TABLES  
          t\_outtab           = it\_vbak  
        EXCEPTIONS  
          program\_error      = 1  
          OTHERS             = 2.  
      IF sy-subrc <> 0.  
\* Implement suitable error handling here  
      ENDIF.  
    ENDIF.  
  ENDIF.

Program#2: Limited Fields with Dictionary Structure

\*&---------------------------------------------------------------------\*  
\*& Report  ZEXAMPLE\_ALV\_01  
\*&  
\*&---------------------------------------------------------------------\*  
\*&  
\*&  
\*&---------------------------------------------------------------------\*  
  
REPORT zexample\_alv\_02.  
TYPES:  
  BEGIN OF ty\_vbak,  
    vbeln TYPE vbeln,  
    erdat TYPE erdat,  
    ernam TYPE ernam,  
    vkorg TYPE vkorg,  
    vtweg TYPE vtweg,  
    spart TYPE spart,  
    kunnr TYPE kunnr,  
    waerk TYPE waerk,  
    netwr TYPE netwr,  
    END OF ty\_vbak,  
    ty\_vbak\_tab TYPE STANDARD TABLE OF ty\_vbak.  
  
DATA: it\_vbak TYPE ty\_vbak\_tab.  
  
  
DATA:  
   lv\_vbeln TYPE vbeln,  
   lv\_kunnr TYPE kunnr.  
PARAMETERS: p\_vkorg TYPE vkorg.  
SELECT-OPTIONS:  
s\_vbeln  FOR lv\_vbeln,  
s\_kunnr FOR lv\_kunnr.  
  
  
START-OF-SELECTION.  
  IF p\_vkorg IS NOT INITIAL  
  OR s\_vbeln IS NOT INITIAL  
  OR s\_kunnr IS NOT INITIAL.  
    SELECT  
      vbeln  
      erdat  
      ernam  
      vkorg  
      vtweg  
      spart  
      kunnr  
      waerk  
      netwr  
      FROM vbak  
      INTO TABLE it\_vbak  
      WHERE vbeln IN s\_vbeln  
      AND vkorg = p\_vkorg  
      AND kunnr IN s\_kunnr.  
    IF it\_vbak IS NOT INITIAL.  
      CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
        EXPORTING  
          i\_callback\_program = sy-repid  
          i\_structure\_name   = 'ZSVBAK\_ALV'  
        TABLES  
          t\_outtab           = it\_vbak  
        EXCEPTIONS  
          program\_error      = 1  
          OTHERS             = 2.  
      IF sy-subrc <> 0.  
\* Implement suitable error handling here  
      ENDIF.  
    ENDIF.  
  ENDIF.

**Output:**

Program#3: Using Field Catalog

Source Code

\*&---------------------------------------------------------------------\*  
\*& Report  ZEXAMPLE\_ALV\_01  
\*&  
\*&---------------------------------------------------------------------\*  
\*&  
\*&  
\*&---------------------------------------------------------------------\*  
  
REPORT zexample\_alv\_03.  
TYPES:  
  BEGIN OF ty\_vbak,  
    vbeln TYPE vbeln,  
    erdat TYPE erdat,  
    ernam TYPE ernam,  
    vkorg TYPE vkorg,  
    vtweg TYPE vtweg,  
    spart TYPE spart,  
    kunnr TYPE kunnr,  
    waerk TYPE waerk,  
    netwr TYPE netwr,  
    END OF ty\_vbak,  
    ty\_vbak\_tab TYPE STANDARD TABLE OF ty\_vbak.  
  
DATA:  
   it\_vbak TYPE ty\_vbak\_tab,  
   it\_fcat TYPE slis\_t\_fieldcat\_alv.  
  
  
DATA:  
   lv\_vbeln TYPE vbeln,  
   lv\_kunnr TYPE kunnr.  
PARAMETERS: p\_vkorg TYPE vkorg.  
SELECT-OPTIONS:  
s\_vbeln  FOR lv\_vbeln,  
s\_kunnr FOR lv\_kunnr.  
  
  
START-OF-SELECTION.  
  IF p\_vkorg IS NOT INITIAL  
  OR s\_vbeln IS NOT INITIAL  
  OR s\_kunnr IS NOT INITIAL.  
    SELECT  
      vbeln  
      erdat  
      ernam  
      vkorg  
      vtweg  
      spart  
      kunnr  
      waerk  
      netwr  
      FROM vbak  
      INTO TABLE it\_vbak  
      WHERE vbeln IN s\_vbeln  
      AND vkorg = p\_vkorg  
      AND kunnr IN s\_kunnr.  
    IF it\_vbak IS NOT INITIAL.  
      " Fill Field Catalog  
      it\_fcat =  VALUE SLIS\_T\_FIELDCAT\_ALV( tabname = 'VBAK'  
      ( fieldname = 'VBELN' seltext\_l = 'Sales Order' seltext\_m = 'Sales Ord' seltext\_s = 'Sls Ord' key = 'X' )  
      ( fieldname = 'ERDAT' seltext\_l = 'Document Date' seltext\_m = 'Doc Date' seltext\_s = 'Doc Dat'  )  
      ( fieldname = 'ERNAM' seltext\_l = 'Created By' seltext\_m = 'Created' seltext\_s = 'Cra By' )  
      ( fieldname = 'VKORG' seltext\_l = 'Sales Org' seltext\_m = 'Sls Org' seltext\_s = 'Sls Org' )  
      ( fieldname = 'VTWEG' seltext\_l = 'Dist Channel' seltext\_m = 'Dist Chan' seltext\_s = 'Dist Cha' )  
      ( fieldname = 'SPART' seltext\_l = 'Division' seltext\_m = 'Division' seltext\_s = 'Div' )  
      ( fieldname = 'KUNNR' seltext\_l = 'Customer' seltext\_m = 'Customer' seltext\_s = 'Cust' )  
      ( fieldname = 'WAERK' seltext\_l = 'Currency' seltext\_m = 'Currency' seltext\_s = 'Curr' )  
      ( fieldname = 'NETWR' seltext\_l = 'Gross Amount' seltext\_m = 'Gross Amnt' seltext\_s = 'Grs Amnt' )  ).  
      CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
        EXPORTING  
          i\_callback\_program = sy-repid  
\*         i\_structure\_name   = 'ZSVBAK\_ALV'  
          it\_fieldcat        = it\_fcat  
        TABLES  
          t\_outtab           = it\_vbak  
        EXCEPTIONS  
          program\_error      = 1  
          OTHERS             = 2.  
      IF sy-subrc <> 0.  
\* Implement suitable error handling here  
      ENDIF.  
    ENDIF.  
  ENDIF.

Program#4: Develop above report in Object Oriented Programming

\*&---------------------------------------------------------------------\*  
\*& Report  ZEXAMPLE\_ALV\_01  
\*&  
\*&---------------------------------------------------------------------\*  
\*&  
\*&  
\*&---------------------------------------------------------------------\*  
  
REPORT zexample\_alv\_04.  
CLASS lcl\_sales DEFINITION.  
  PUBLIC SECTION.  
  " Sales Declaration  
    TYPES:  
      BEGIN OF ty\_vbak,  
        vbeln TYPE vbeln,  
        erdat TYPE erdat,  
        ernam TYPE ernam,  
        vkorg TYPE vkorg,  
        vtweg TYPE vtweg,  
        spart TYPE spart,  
        kunnr TYPE kunnr,  
        waerk TYPE waerk,  
        netwr TYPE netwr,  
        END OF ty\_vbak,  
        ty\_vbak\_tab TYPE STANDARD TABLE OF ty\_vbak.  
    TYPES:  
      ty\_kunnr\_r TYPE RANGE OF kunnr,  
      ty\_vbeln\_r TYPE RANGE OF vbeln.  
  
    METHODS: constructor IMPORTING  iv\_vkorg TYPE vkorg  
                                    ir\_vbeln TYPE ty\_vbeln\_r  
                                    ir\_kunnr TYPE ty\_kunnr\_r,  
             get\_data RETURNING VALUE(rv\_okay) TYPE char1,  
             display.  
  PROTECTED SECTION.  
  PRIVATE SECTION.  
    DATA:  
       g\_vkorg TYPE vkorg,  
       r\_kunnr TYPE RANGE OF kunnr,  
       r\_vbeln TYPE RANGE OF vbeln,  
       it\_vbak TYPE ty\_vbak\_tab,  
       it\_fcat TYPE slis\_t\_fieldcat\_alv.  
    METHODS:  
     build\_fcat RETURNING VALUE(rt\_fcat) TYPE slis\_t\_fieldcat\_alv.  
ENDCLASS.  
CLASS lcl\_sales IMPLEMENTATION.  
\*--------------------------------------------------------------------\*  
\*  
\*--------------------------------------------------------------------\*  
  METHOD constructor.  
    g\_vkorg = iv\_vkorg.  
    r\_vbeln = ir\_vbeln.  
    r\_kunnr = ir\_kunnr.  
  ENDMETHOD.  
\*--------------------------------------------------------------------\*  
\*  
\*--------------------------------------------------------------------\*  
  METHOD get\_data.  
    SELECT  
      vbeln  
      erdat  
      ernam  
      vkorg  
      vtweg  
      spart  
      kunnr  
      waerk  
      netwr  
      FROM vbak  
      INTO TABLE it\_vbak  
      WHERE vbeln IN r\_vbeln  
      AND vkorg = g\_vkorg  
      AND kunnr IN r\_kunnr.  
    IF sy-subrc = 0.  
      rv\_okay = 'X'.  
    ELSE.  
      CLEAR rv\_okay.  
    ENDIF.  
  ENDMETHOD.  
\*--------------------------------------------------------------------\*  
\*  
\*--------------------------------------------------------------------\*  
  METHOD build\_fcat.  
    " Fill Field Catalog  
    rt\_fcat =  VALUE slis\_t\_fieldcat\_alv( tabname = 'VBAK'  
    ( fieldname = 'VBELN' seltext\_l = 'Sales Order' seltext\_m = 'Sales Ord' seltext\_s = 'Sls Ord' key = 'X' )  
    ( fieldname = 'ERDAT' seltext\_l = 'Document Date' seltext\_m = 'Doc Date' seltext\_s = 'Doc Dat'  )  
    ( fieldname = 'ERNAM' seltext\_l = 'Created By' seltext\_m = 'Created' seltext\_s = 'Cra By' )  
    ( fieldname = 'VKORG' seltext\_l = 'Sales Org' seltext\_m = 'Sls Org' seltext\_s = 'Sls Org' )  
    ( fieldname = 'VTWEG' seltext\_l = 'Dist Channel' seltext\_m = 'Dist Chan' seltext\_s = 'Dist Cha' )  
    ( fieldname = 'SPART' seltext\_l = 'Division' seltext\_m = 'Division' seltext\_s = 'Div' )  
    ( fieldname = 'KUNNR' seltext\_l = 'Customer' seltext\_m = 'Customer' seltext\_s = 'Cust' )  
    ( fieldname = 'WAERK' seltext\_l = 'Currency' seltext\_m = 'Currency' seltext\_s = 'Curr' )  
    ( fieldname = 'NETWR' seltext\_l = 'Gross Amount' seltext\_m = 'Gross Amnt' seltext\_s = 'Grs Amnt' )  ).  
  ENDMETHOD.  
\*--------------------------------------------------------------------\*  
\*--------------------------------------------------------------------\*  
  METHOD display.  
  
    CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
      EXPORTING  
        i\_callback\_program = sy-repid  
\*       i\_structure\_name   = 'ZSVBAK\_ALV'  
        it\_fieldcat        = build\_fcat( ) "it\_fcat  
      TABLES  
        t\_outtab           = it\_vbak  
      EXCEPTIONS  
        program\_error      = 1  
        OTHERS             = 2.  
    IF sy-subrc <> 0.  
\* Implement suitable error handling here  
    ENDIF.  
  ENDMETHOD.  
ENDCLASS.  
  
\* Main Program  
DATA:  
   lv\_vbeln TYPE vbeln,  
   lv\_kunnr TYPE kunnr.  
PARAMETERS: p\_vkorg TYPE vkorg.  
SELECT-OPTIONS:  
s\_vbeln  FOR lv\_vbeln,  
s\_kunnr FOR lv\_kunnr.  
  
START-OF-SELECTION.  
  DATA(lo\_sales) = NEW lcl\_sales(  
                        iv\_vkorg = p\_vkorg  
                        ir\_vbeln = s\_vbeln[]  
                        ir\_kunnr = s\_kunnr[] ).  
  IF lo\_sales IS BOUND.  
    IF lo\_sales->get\_data( ) IS NOT INITIAL.  
      lo\_sales->display( ).  
    ENDIF.  
  ENDIF.

Program#5: Handling Double Click

*\*&---------------------------------------------------------------------\**  
*\*& Report  ZEXAMPLE\_ALV\_01*  
*\*&*  
*\*&---------------------------------------------------------------------\**  
*\*&*  
*\*&*  
*\*&---------------------------------------------------------------------\**  
  
REPORT zexample\_alv\_05.  
TYPES:  
  BEGIN OF ty\_vbak,  
    vbeln TYPE vbeln,  
    erdat TYPE erdat,  
    ernam TYPE ernam,  
    vkorg TYPE vkorg,  
    vtweg TYPE vtweg,  
    spart TYPE spart,  
    kunnr TYPE kunnr,  
    waerk TYPE waerk,  
    netwr TYPE netwr,  
    END OF ty\_vbak,  
    ty\_vbak\_tab TYPE STANDARD TABLE OF ty\_vbak.  
  
DATA:  
   it\_vbak TYPE ty\_vbak\_tab.  
*\*   it\_fcat TYPE slis\_t\_fieldcat\_alv.*  
  
  
DATA:  
   lv\_vbeln TYPE vbeln,  
   lv\_kunnr TYPE kunnr.  
PARAMETERS: p\_vkorg TYPE vkorg.  
SELECT-OPTIONS:  
s\_vbeln  FOR lv\_vbeln,  
s\_kunnr FOR lv\_kunnr.  
  
  
START-OF-SELECTION.  
  
  IF p\_vkorg IS NOT INITIAL  
  OR s\_vbeln IS NOT INITIAL  
  OR s\_kunnr IS NOT INITIAL.  
    SELECT  
      vbeln  
      erdat  
      ernam  
      vkorg  
      vtweg  
      spart  
      kunnr  
      waerk  
      netwr  
      FROM vbak  
      INTO TABLE it\_vbak  
      WHERE vbeln IN s\_vbeln  
      AND vkorg = p\_vkorg  
      AND kunnr IN s\_kunnr.  
    IF it\_vbak IS NOT INITIAL.  
      *" Fill Field Catalog*  
      DATA(lt\_fcat) =  VALUE slis\_t\_fieldcat\_alv( tabname = 'VBAK'  
      ( fieldname = 'VBELN' seltext\_l = 'Sales Order' seltext\_m = 'Sales Ord' seltext\_s = 'Sls Ord' key = 'X' )  
      ( fieldname = 'ERDAT' seltext\_l = 'Document Date' seltext\_m = 'Doc Date' seltext\_s = 'Doc Dat'  )  
      ( fieldname = 'ERNAM' seltext\_l = 'Created By' seltext\_m = 'Created' seltext\_s = 'Cra By' )  
      ( fieldname = 'VKORG' seltext\_l = 'Sales Org' seltext\_m = 'Sls Org' seltext\_s = 'Sls Org' )  
      ( fieldname = 'VTWEG' seltext\_l = 'Dist Channel' seltext\_m = 'Dist Chan' seltext\_s = 'Dist Cha' )  
      ( fieldname = 'SPART' seltext\_l = 'Division' seltext\_m = 'Division' seltext\_s = 'Div' )  
      ( fieldname = 'KUNNR' seltext\_l = 'Customer' seltext\_m = 'Customer' seltext\_s = 'Cust' )  
      ( fieldname = 'WAERK' seltext\_l = 'Currency' seltext\_m = 'Currency' seltext\_s = 'Curr' )  
      ( fieldname = 'NETWR' seltext\_l = 'Gross Amount' seltext\_m = 'Gross Amnt' seltext\_s = 'Grs Amnt' )  ).  
      CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
        EXPORTING  
          i\_callback\_program      = sy-repid  
*\*         i\_structure\_name        = 'ZSVBAK\_ALV'*  
          it\_fieldcat             = lt\_fcat  
          i\_callback\_user\_command = 'HANDLE\_USERCOMMAND'  
        TABLES  
          t\_outtab                = it\_vbak  
        EXCEPTIONS  
          program\_error           = 1  
          OTHERS                  = 2.  
      IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
FORM handle\_usercommand USING r\_ucomm LIKE sy-ucomm  
                              rs\_selfield TYPE slis\_selfield.  
  TYPES:  
    BEGIN OF ty\_vbap,  
      vbeln TYPE vbeln,  
      posnr TYPE posnr,  
      matnr TYPE matnr,  
      zmeng TYPE vbap-zmeng,  
    END OF ty\_vbap,  
    ty\_vbap\_tab TYPE STANDARD TABLE OF ty\_vbap.  
  DATA lt\_vbap TYPE ty\_vbap\_tab.  
  CASE r\_ucomm.  
    WHEN '&IC1'. *" Double Click*  
      *" Get selected record from IT\_VBAK*  
      TRY.  
          DATA(lv\_vbeln) = it\_vbak[ rs\_selfield-tabindex ]-vbeln.  
          *" Get Data*  
          SELECT vbeln posnr matnr zmeng  
            FROM vbap  
            INTO TABLE lt\_vbap  
            WHERE vbeln = lv\_vbeln.  
          *" Build Fcat*  
          DATA(lt\_fcat) = VALUE slis\_t\_fieldcat\_alv( tabname = 'VBAP'  
                ( fieldname = 'VBELN' seltext\_l = 'Sales Order' seltext\_m = 'Sales Ord' seltext\_s = 'Sls Ord' key = 'X' )  
                ( fieldname = 'POSNR' seltext\_l = 'Item' seltext\_m = 'Item' seltext\_s = 'Item' key = 'X' )  
                ( fieldname = 'MATNR' seltext\_l = 'Product' seltext\_m = 'Product' seltext\_s = 'Product' )  
                ( fieldname = 'ZMENG' seltext\_l = 'Quantity' seltext\_m = 'Quantity' seltext\_s = 'Quantity' ) ).  
          *" Display ALV*  
          CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
            EXPORTING  
              i\_callback\_program = sy-repid  
              it\_fieldcat        = lt\_fcat  
            TABLES  
              t\_outtab           = lt\_vbap  
            EXCEPTIONS  
              program\_error      = 1  
              OTHERS             = 2.  
          IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
          ENDIF.  
        CATCH cx\_sy\_itab\_line\_not\_found.  
      ENDTRY.  
  ENDCASE.  
ENDFORM.

Program# Set PF Status and Handling the Button Click

*\*&---------------------------------------------------------------------\**  
*\*& Report  ZEXAMPLE\_ALV\_01*  
*\*&*  
*\*&---------------------------------------------------------------------\**  
*\*&*  
*\*&*  
*\*&---------------------------------------------------------------------\**  
  
REPORT zexample\_alv\_06.  
TYPES:  
  BEGIN OF ty\_vbak,  
    vbeln TYPE vbeln,  
    erdat TYPE erdat,  
    ernam TYPE ernam,  
    vkorg TYPE vkorg,  
    vtweg TYPE vtweg,  
    spart TYPE spart,  
    kunnr TYPE kunnr,  
    waerk TYPE waerk,  
    netwr TYPE netwr,  
    END OF ty\_vbak,  
    ty\_vbak\_tab TYPE STANDARD TABLE OF ty\_vbak.  
  
DATA:  
   it\_vbak TYPE ty\_vbak\_tab.  
*\*   it\_fcat TYPE slis\_t\_fieldcat\_alv.*  
  
  
DATA:  
   lv\_vbeln TYPE vbeln,  
   lv\_kunnr TYPE kunnr.  
PARAMETERS: p\_vkorg TYPE vkorg.  
SELECT-OPTIONS:  
s\_vbeln  FOR lv\_vbeln,  
s\_kunnr FOR lv\_kunnr.  
  
  
START-OF-SELECTION.  
  
  IF p\_vkorg IS NOT INITIAL  
  OR s\_vbeln IS NOT INITIAL  
  OR s\_kunnr IS NOT INITIAL.  
    SELECT  
      vbeln  
      erdat  
      ernam  
      vkorg  
      vtweg  
      spart  
      kunnr  
      waerk  
      netwr  
      FROM vbak  
      INTO TABLE it\_vbak  
      WHERE vbeln IN s\_vbeln  
      AND vkorg = p\_vkorg  
      AND kunnr IN s\_kunnr.  
    IF it\_vbak IS NOT INITIAL.  
      *" Fill Field Catalog*  
      DATA(lt\_fcat) =  VALUE slis\_t\_fieldcat\_alv( tabname = 'VBAK'  
      ( fieldname = 'VBELN' seltext\_l = 'Sales Order' seltext\_m = 'Sales Ord' seltext\_s = 'Sls Ord' key = 'X' )  
      ( fieldname = 'ERDAT' seltext\_l = 'Document Date' seltext\_m = 'Doc Date' seltext\_s = 'Doc Dat'  )  
      ( fieldname = 'ERNAM' seltext\_l = 'Created By' seltext\_m = 'Created' seltext\_s = 'Cra By' )  
      ( fieldname = 'VKORG' seltext\_l = 'Sales Org' seltext\_m = 'Sls Org' seltext\_s = 'Sls Org' )  
      ( fieldname = 'VTWEG' seltext\_l = 'Dist Channel' seltext\_m = 'Dist Chan' seltext\_s = 'Dist Cha' )  
      ( fieldname = 'SPART' seltext\_l = 'Division' seltext\_m = 'Division' seltext\_s = 'Div' )  
      ( fieldname = 'KUNNR' seltext\_l = 'Customer' seltext\_m = 'Customer' seltext\_s = 'Cust' )  
      ( fieldname = 'WAERK' seltext\_l = 'Currency' seltext\_m = 'Currency' seltext\_s = 'Curr' )  
      ( fieldname = 'NETWR' seltext\_l = 'Gross Amount' seltext\_m = 'Gross Amnt' seltext\_s = 'Grs Amnt' )  ).  
      CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
        EXPORTING  
          i\_callback\_program       = sy-repid  
*\*         i\_structure\_name         = 'ZSVBAK\_ALV'*  
          it\_fieldcat              = lt\_fcat  
          i\_callback\_user\_command  = 'HANDLE\_USERCOMMAND'  
          i\_callback\_pf\_status\_set = 'SET\_PFSTATUS'  
        TABLES  
          t\_outtab                 = it\_vbak  
        EXCEPTIONS  
          program\_error            = 1  
          OTHERS                   = 2.  
      IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
      ENDIF.  
    ENDIF.  
  ENDIF.  
  
FORM handle\_usercommand USING r\_ucomm LIKE sy-ucomm  
                              rs\_selfield TYPE slis\_selfield.  
  TYPES:  
    BEGIN OF ty\_vbap,  
      vbeln TYPE vbeln,  
      posnr TYPE posnr,  
      matnr TYPE matnr,  
      zmeng TYPE vbap-zmeng,  
    END OF ty\_vbap,  
    ty\_vbap\_tab TYPE STANDARD TABLE OF ty\_vbap.  
  DATA lt\_vbap TYPE ty\_vbap\_tab.  
  CASE r\_ucomm.  
    WHEN '&IC1' OR  *" Double Click*  
         'DISP'.    *" Button Clicked*  
      *" Get selected record from IT\_VBAK*  
      TRY.  
          DATA(lv\_vbeln) = it\_vbak[ rs\_selfield-tabindex ]-vbeln.  
          *" Get Data*  
          SELECT vbeln posnr matnr zmeng  
            FROM vbap  
            INTO TABLE lt\_vbap  
            WHERE vbeln = lv\_vbeln.  
          *" Build Fcat*  
          DATA(lt\_fcat) = VALUE slis\_t\_fieldcat\_alv( tabname = 'VBAP'  
                ( fieldname = 'VBELN' seltext\_l = 'Sales Order' seltext\_m = 'Sales Ord' seltext\_s = 'Sls Ord' key = 'X' )  
                ( fieldname = 'POSNR' seltext\_l = 'Item' seltext\_m = 'Item' seltext\_s = 'Item' key = 'X' )  
                ( fieldname = 'MATNR' seltext\_l = 'Product' seltext\_m = 'Product' seltext\_s = 'Product' )  
                ( fieldname = 'ZMENG' seltext\_l = 'Quantity' seltext\_m = 'Quantity' seltext\_s = 'Quantity' ) ).  
          *" Display ALV*  
          CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
            EXPORTING  
              i\_callback\_program = sy-repid  
              it\_fieldcat        = lt\_fcat  
            TABLES  
              t\_outtab           = lt\_vbap  
            EXCEPTIONS  
              program\_error      = 1  
              OTHERS             = 2.  
          IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
          ENDIF.  
        CATCH cx\_sy\_itab\_line\_not\_found.  
      ENDTRY.  
  ENDCASE.  
ENDFORM.  
  
  
FORM set\_pfstatus USING rt\_extab TYPE slis\_t\_extab.  
  SET PF-STATUS 'STATUS1'.  
ENDFORM.

OOPS ALV

1. Without Custom Container
2. With Custom Container
3. Handling Double Click
4. Adding Buttons and Handling Button Click