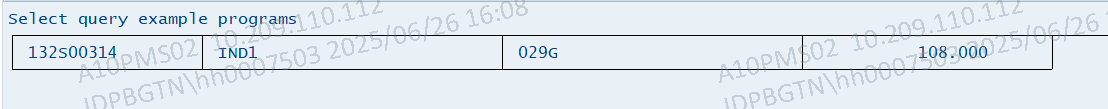
Database tables query and updates

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Contents** | **Page NO** |
| 1.1 | Select single |  |
| 1.2 | Select 4 fields |  |
| 1.2.1 | Select 4 fields where in additions |  |
| 1.2.2 | Select 4 fields where in & and additions |  |
| 1.2.3 | Select 4 fields where in & or additions |  |
| 1.3 | Select all fields |  |
| 1.3.1 | Select corresponding fields |  |
| 1.4 | Ranges and select options |  |
| 1.4.1 | Select options – No extension |  |
| 1.4.2 | Select options set initial ranges |  |
| 1.5 | Select use the package size |  |
| 1.6 | Select query using order by and up to rows |  |
| 1.7 | Inner join 2 tables |  |
| 1.7.1 | Inner join 2 more tables |  |
| 1.7.2 | Left outer join |  |
| 1.8 | For all entries |  |
| 1.9.1 | Modify DB |  |
| 1.9.2 | Insert DB |  |
| 1.9.3 | Update DB |  |
| 1.9.4 | Delete DB |  |
| 1.10 | How to find number of columns in database table |  |
| 1.11 | Secondary index use in select query |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

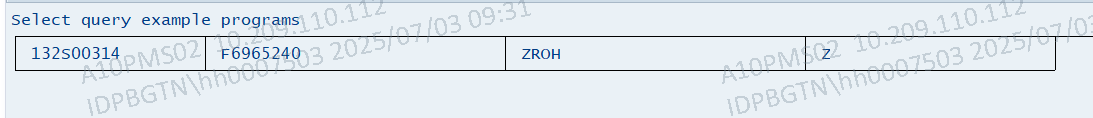
1.1 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_MARD,  
         MATNR LIKE MARD-MATNR,  
         WERKS LIKE MARD-WERKS,  
         LGORT LIKE MARD-LGORT,  
         LABST LIKE MARD-LABST,  
       END OF TY\_MARD.  
  
DATA: LS\_MARD TYPE TY\_MARD.  
  
SELECT SINGLE MATNR WERKS LGORT LABST FROM MARD INTO LS\_MARD WHERE MATNR = '132S00314'.  
  
WRITE: / SY-VLINE, LS\_MARD-MATNR,  
       20 SY-VLINE, LS\_MARD-WERKS,  
       50 SY-VLINE, LS\_MARD-LGORT,  
       80 SY-VLINE, LS\_MARD-LABST,  
       105 SY-VLINE.  
ULINE AT (105).

Output



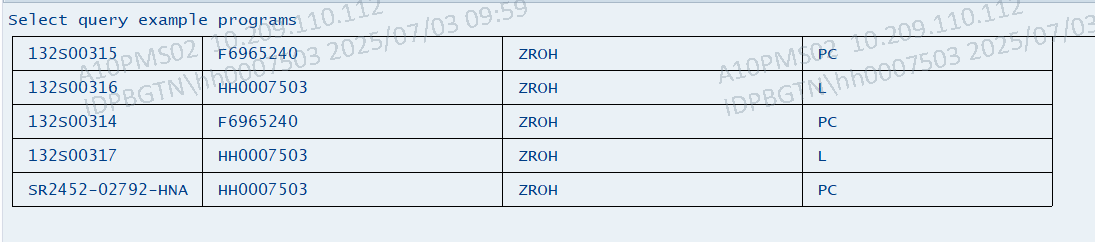
1.2 TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MBRSH TYPE MBRSH,  
       END OF TY\_MARA.  
  
DATA: LT\_MARD TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT MATNR ERNAM MTART MBRSH FROM MARA INTO TABLE LT\_MARD WHERE MATNR = '132S00314'.  
  
LOOP AT LT\_MARD INTO LS\_MARA.  
  WRITE: / SY-VLINE, LS\_MARA-MATNR,  
         20 SY-VLINE, LS\_MARA-ERNAM,  
         50 SY-VLINE, LS\_MARA-MTART,  
         80 SY-VLINE, LS\_MARA-MBRSH,  
         105 SY-VLINE.  
  ULINE AT (105).  
ENDLOOP.

Output



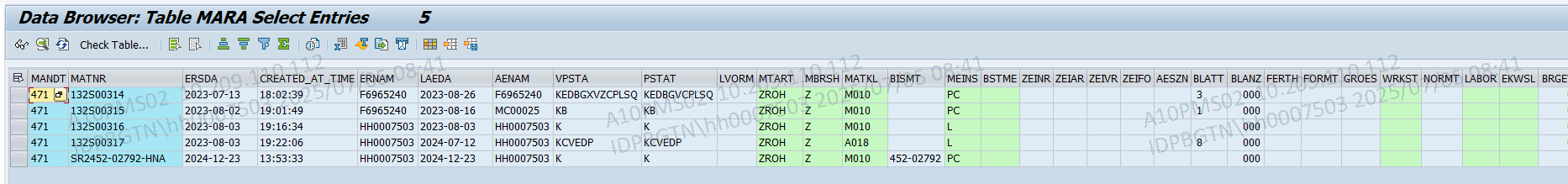
1.2.1 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARD TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT MATNR ERNAM MTART MEINS FROM MARA INTO TABLE LT\_MARD WHERE MEINS IN ( 'PC', 'L' ).  
  
LOOP AT LT\_MARD INTO LS\_MARA.  
  WRITE: / SY-VLINE, LS\_MARA-MATNR,  
         20 SY-VLINE, LS\_MARA-ERNAM,  
         50 SY-VLINE, LS\_MARA-MTART,  
         80 SY-VLINE, LS\_MARA-MEINS,  
         105 SY-VLINE.  
  ULINE AT (105).  
ENDLOOP.

Output

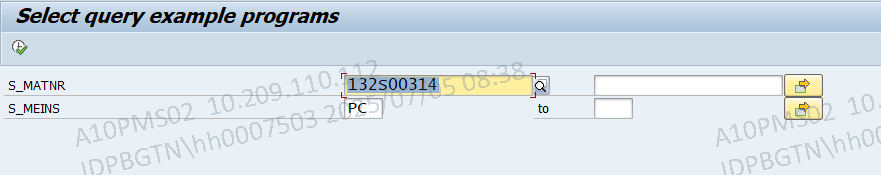


1.2.2 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARD TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT-OPTIONS: S\_MATNR FOR MARA-MATNR,  
                S\_MEINS FOR MARA-MEINS.  
  
*\*SELECT MATNR ERNAM MTART MEINS*  
*\*FROM MARA*  
*\*INTO TABLE LT\_MARD*  
*\*WHERE MATNR IN S\_MATNR OR MEINS IN S\_MEINS.*  
*" IF SELECT OPTION FILL IS EMPTY IT WILL SELECT ALL DATA.*  
  
SELECT MATNR ERNAM MTART MEINS  
FROM MARA  
INTO TABLE LT\_MARD  
WHERE MATNR IN S\_MATNR AND MEINS IN S\_MEINS.  
  
LOOP AT LT\_MARD INTO LS\_MARA.  
  WRITE: /   SY-VLINE, LS\_MARA-MATNR,  
         20  SY-VLINE, LS\_MARA-ERNAM,  
         50  SY-VLINE, LS\_MARA-MTART,  
         80  SY-VLINE, LS\_MARA-MEINS,  
         105 SY-VLINE.  
  ULINE AT (105).  
ENDLOOP.

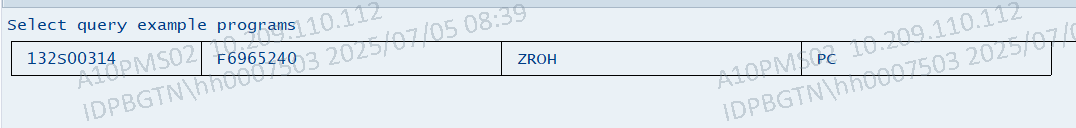
DB



Input

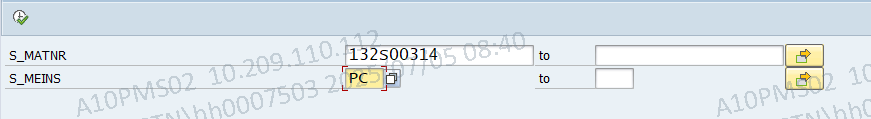


Output

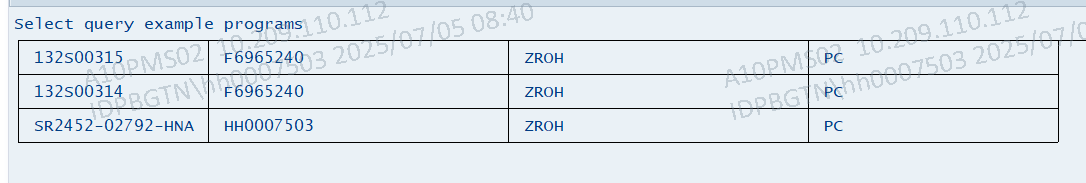


1.2.3 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARD TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT-OPTIONS: S\_MATNR FOR MARA-MATNR,  
                S\_MEINS FOR MARA-MEINS.  
  
SELECT MATNR ERNAM MTART MEINS  
FROM MARA  
INTO TABLE LT\_MARD  
WHERE MATNR IN S\_MATNR OR MEINS IN S\_MEINS.  
*\* IF SELECT OPTION FILL IS EMPTY IT WILL SELECT ALL DATA.*  
  
*\*SELECT MATNR ERNAM MTART MEINS*  
*\*FROM MARA*  
*\*INTO TABLE LT\_MARD*  
*\*WHERE MATNR IN S\_MATNR AND MEINS IN S\_MEINS.*  
  
LOOP AT LT\_MARD INTO LS\_MARA.  
  WRITE: /   SY-VLINE, LS\_MARA-MATNR,  
         20  SY-VLINE, LS\_MARA-ERNAM,  
         50  SY-VLINE, LS\_MARA-MTART,  
         80  SY-VLINE, LS\_MARA-MEINS,  
         105 SY-VLINE.  
  ULINE AT (105).  
ENDLOOP.

Input

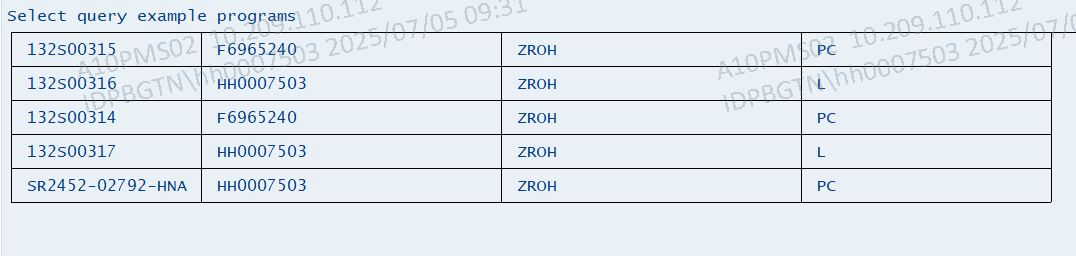


Output



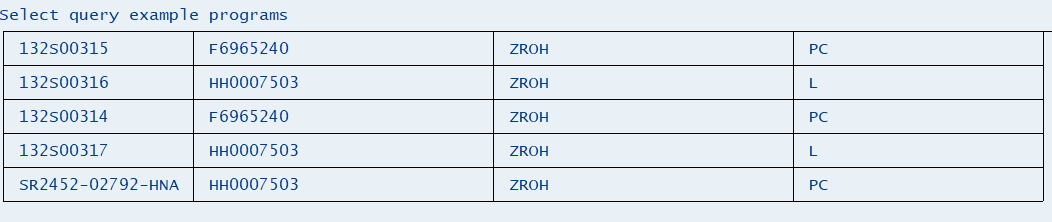
1.3 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
DATA: LT\_MARA TYPE STANDARD TABLE OF MARA,  
      LS\_MARA TYPE MARA.  
  
  
SELECT \* FROM MARA INTO TABLE LT\_MARA.  
  
LOOP AT LT\_MARA INTO LS\_MARA.  
  WRITE: /   SY-VLINE, LS\_MARA-MATNR,  
          20  SY-VLINE, LS\_MARA-ERNAM,  
          50  SY-VLINE, LS\_MARA-MTART,  
          80  SY-VLINE, LS\_MARA-MEINS,  
          105 SY-VLINE.  
  ULINE AT (105).  
ENDLOOP.

Output



1.3.1 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARA TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT \* FROM MARA INTO CORRESPONDING FIELDS OF TABLE LT\_MARA.  
*" CORRESPONDING FIELS SELECT THE FIELDS CORRECTLY EVEN ALSO SELECT ALL FIELDS*  
  
LOOP AT LT\_MARA INTO LS\_MARA.  
  WRITE: /   SY-VLINE, LS\_MARA-MATNR,  
          20  SY-VLINE, LS\_MARA-ERNAM,  
          50  SY-VLINE, LS\_MARA-MTART,  
          80  SY-VLINE, LS\_MARA-MEINS,  
          105 SY-VLINE.  
  ULINE AT (105).  
ENDLOOP.

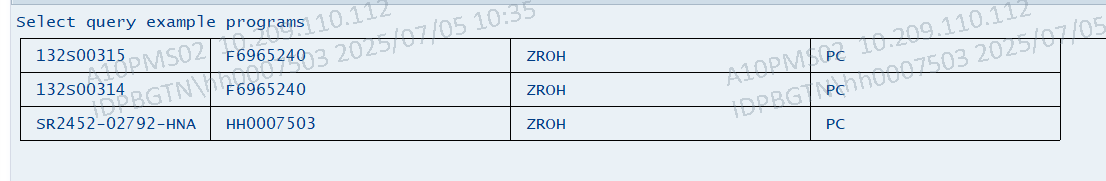
Output



1.4 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARA TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT-OPTIONS S\_MATNR FOR MARA-MATNR.  
RANGES R\_MTART FOR MARA-MATNR.  
RANGES R\_MEINS FOR MARA-MEINS.  
  
START-OF-SELECTION.  
  
  SELECT MATNR ERNAM MTART MEINS  
  FROM MARA  
  INTO TABLE LT\_MARA  
  WHERE MATNR IN S\_MATNR AND MTART IN R\_MTART AND MEINS IN R\_MEINS.  
  
  LOOP AT LT\_MARA INTO LS\_MARA.  
    WRITE: /   SY-VLINE, LS\_MARA-MATNR,  
            20  SY-VLINE, LS\_MARA-ERNAM,  
            50  SY-VLINE, LS\_MARA-MTART,  
            80  SY-VLINE, LS\_MARA-MEINS,  
            105 SY-VLINE.  
    ULINE AT (105).  
  ENDLOOP.

INITIALIZATION.  
  R\_MTART-SIGN   = 'I'.  
  R\_MTART-OPTION = 'EQ'.  
  R\_MTART-LOW    = 'ZROH'.  
  APPEND R\_MTART.  
  CLEAR R\_MTART.  
  
  R\_MEINS-SIGN   = 'I'.  
  R\_MEINS-OPTION = 'EQ'.  
  R\_MEINS-LOW    = 'PC'.  
  APPEND R\_MEINS.  
  CLEAR R\_MEINS.

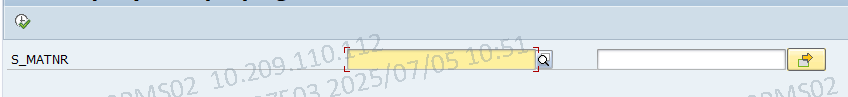
Output



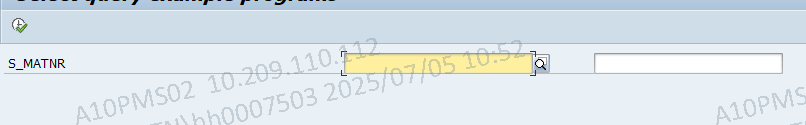
1.4.1 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARA TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT-OPTIONS S\_MATNR FOR MARA-MATNR NO-EXTENSION.  
RANGES R\_MTART FOR MARA-MATNR.  
RANGES R\_MEINS FOR MARA-MEINS.  
  
START-OF-SELECTION.  
  
  SELECT MATNR ERNAM MTART MEINS  
  FROM MARA  
  INTO TABLE LT\_MARA  
  WHERE MATNR IN S\_MATNR AND MTART IN R\_MTART AND MEINS IN R\_MEINS.  
  
  LOOP AT LT\_MARA INTO LS\_MARA.  
    WRITE: /    SY-VLINE, LS\_MARA-MATNR,  
            20  SY-VLINE, LS\_MARA-ERNAM,  
            50  SY-VLINE, LS\_MARA-MTART,  
            80  SY-VLINE, LS\_MARA-MEINS,  
            105 SY-VLINE.  
    ULINE AT (105).  
  ENDLOOP.

INITIALIZATION.  
  R\_MTART-SIGN   = 'I'.  
  R\_MTART-OPTION = 'EQ'.  
  R\_MTART-LOW    = 'ZROH'.  
  APPEND R\_MTART.  
  CLEAR R\_MTART.  
  
  R\_MEINS-SIGN   = 'I'.  
  R\_MEINS-OPTION = 'EQ'.  
  R\_MEINS-LOW    = 'PC'.  
  APPEND R\_MEINS.  
  CLEAR R\_MEINS.

Output with extension



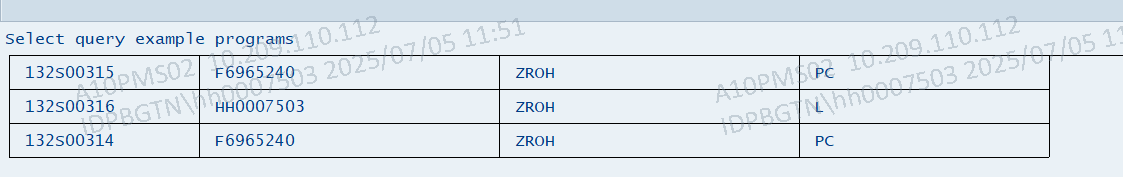
Output



1.4.2 REPORT ZYP\_SELECT\_QUERY.  
  
TABLES MARA.  
  
TYPES: BEGIN OF TY\_MARA,  
         MATNR TYPE MATNR,  
         ERNAM TYPE ERNAM,  
         MTART TYPE MTART,  
         MEINS TYPE MEINS,  
       END OF TY\_MARA.  
  
DATA: LT\_MARA TYPE STANDARD TABLE OF TY\_MARA,  
      LS\_MARA TYPE TY\_MARA.  
  
SELECT-OPTIONS S\_MATNR FOR MARA-MATNR NO-EXTENSION.  
RANGES R\_MTART FOR MARA-MATNR.  
RANGES R\_MEINS FOR MARA-MEINS.  
  
START-OF-SELECTION.  
  
  SELECT MATNR ERNAM MTART MEINS  
  FROM MARA  
  INTO TABLE LT\_MARA  
  WHERE MATNR IN S\_MATNR.  
  
  LOOP AT LT\_MARA INTO LS\_MARA.  
    WRITE: /   SY-VLINE, LS\_MARA-MATNR,  
            20  SY-VLINE, LS\_MARA-ERNAM,  
            50  SY-VLINE, LS\_MARA-MTART,  
            80  SY-VLINE, LS\_MARA-MEINS,  
            105 SY-VLINE.  
    ULINE AT (105).  
  ENDLOOP.

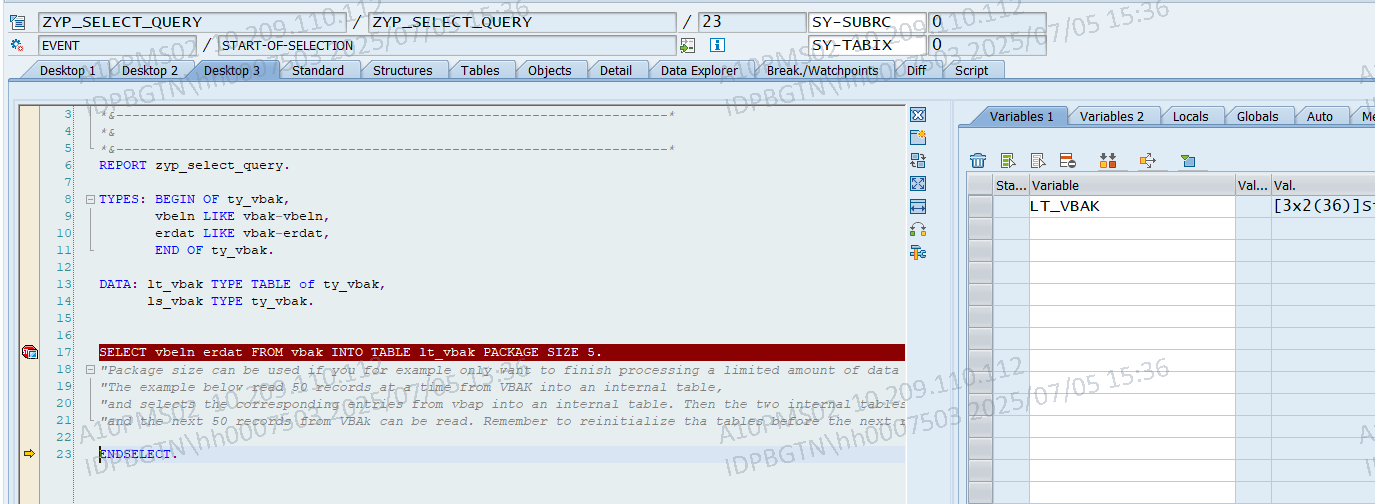
INITIALIZATION.  
  S\_MATNR-SIGN   = 'I'.  
  S\_MATNR-OPTION = 'BT'.  
  S\_MATNR-LOW    = '132S00314'.  
  S\_MATNR-HIGH   = '132S00316'.  
  APPEND S\_MATNR.  
  CLEAR S\_MATNR.

Output



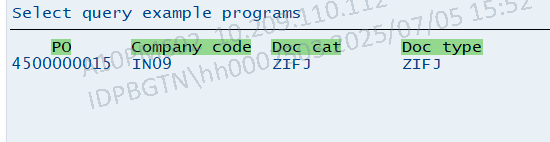
1.5 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_VBAK,  
       VBELN LIKE VBAK-VBELN,  
       ERDAT LIKE VBAK-ERDAT,  
       END OF TY\_VBAK.  
  
DATA: LT\_VBAK TYPE TABLE OF TY\_VBAK,  
      LS\_VBAK TYPE TY\_VBAK.  
  
  
SELECT VBELN ERDAT FROM VBAK INTO TABLE LT\_VBAK PACKAGE SIZE 5.  
*"PACKAGE SIZE CAN BE USED IF YOU FOR EXAMPLE ONLY WANT TO FINISH PROCESSING A LIMITED AMOUNT OF DATA AT A TIME DUE TO LACK OF MEMORY.THE EXAMPLE BELOW READ 50 RECORDS AT A TIME FROM VBAK INTO AN INTERNAL TABLE, AND SELECTS THE CORRESPONDING ENTRIES FROM VBAP INTO AN INTERNAL TABLE. THEN THE TWO INTERNAL TABLES CAN BE PROCESSED,AND THE NEXT 50 RECORDS FROM VBAK CAN BE READ. REMEMBER TO REINITIALIZE THA TABLES BEFORE THE NEXT READ.*  
  
ENDSELECT.

Output



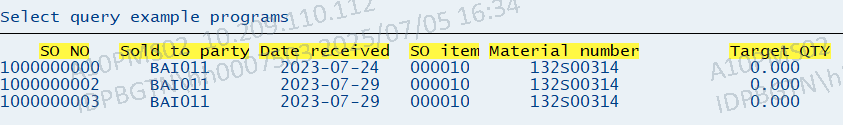
1.6 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_EKKO,  
         EBELN TYPE EBELN,  
         BUKRS TYPE BUKRS,  
         BSTYP TYPE BSTYP,  
         BSART TYPE BSART,  
       END OF TY\_EKKO.  
  
DATA: LT\_EKKO TYPE STANDARD TABLE OF TY\_EKKO,  
      LS\_EKKO TYPE TY\_EKKO.  
  
SELECT EBELN BUKRS BSTYP BSART UP TO 1 ROWS FROM EKKO INTO TABLE LT\_EKKO WHERE EBELN = '4500000015' ORDER BY BUKRS DESCENDING.  
  
WRITE:/5 'PO' COLOR 5,13 'COMPANY CODE' COLOR 5, 27 'DOC CAT' COLOR 5,40 'DOC TYPE' COLOR 5.  
LOOP AT LT\_EKKO INTO LS\_EKKO.  
  WRITE: / LS\_EKKO-EBELN,  
          13 LS\_EKKO-BUKRS,  
          27 LS\_EKKO-BSART,  
          40 LS\_EKKO-BSART.  
ENDLOOP.

Output



1.7 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_SO,  
         VBELN TYPE VBELN,  
         KUNNR TYPE KUNNR,  
         AUDAT TYPE AUDAT,  
         POSNR TYPE POSNR,  
         MATNR TYPE MATNR,  
         ZMENG TYPE VBAP-ZMENG,  
       END OF TY\_SO.  
  
DATA: LT\_SO TYPE STANDARD TABLE OF TY\_SO,  
      LS\_SO TYPE TY\_SO.  
  
START-OF-SELECTION.  
  
  SELECT VBAK~VBELN VBAK~KUNNR VBAK~AUDAT VBAP~POSNR VBAP~MATNR VBAP~ZMENG  
  INTO TABLE LT\_SO  
  FROM VBAK  
  INNER JOIN VBAP ON VBAK~VBELN = VBAP~VBELN.  
  
  WRITE: 5 'SO NO' COLOR 3, 13 'SOLD TO PARTY' COLOR 3, 27 'DATE RECEIVED' COLOR 3,42 'SO ITEM' COLOR 3, 'MATERIAL NUMBER' COLOR 3,74 'TARGET QTY' COLOR 3.  
  
  
  LOOP AT LT\_SO INTO LS\_SO.  
    WRITE: / LS\_SO-VBELN,  
            16 LS\_SO-KUNNR,  
            29 LS\_SO-AUDAT,  
            42 LS\_SO-POSNR,  
            54 LS\_SO-MATNR,  
            65 LS\_SO-ZMENG.  
  ENDLOOP.

Output

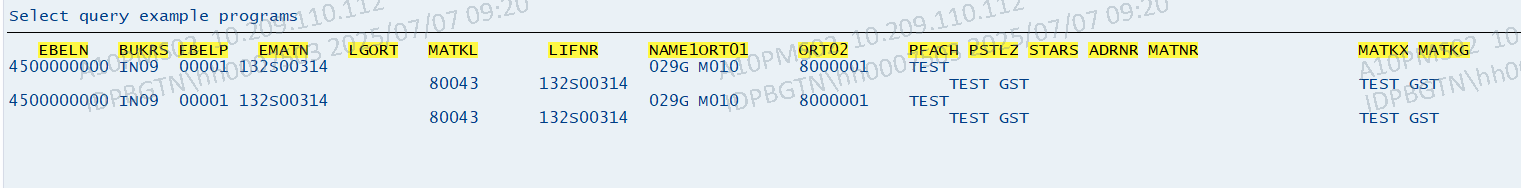


1.7.1 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_PO,  
         EBELN TYPE EBELN,  
         BUKRS TYPE BUKRS,  
         EBELP TYPE EBELP,  
         EMATN TYPE EMATN,  
         LGORT TYPE EKPO-LGORT,  
         MATKL TYPE MATKL,  
         LIFNR TYPE LIFNR,  
         NAME1 TYPE NAME1,  
         ORT01 TYPE ORT01,  
         ORT02 TYPE ORT02,  
         PFACH TYPE PFACH,  
         PSTLZ TYPE PSTLZ,  
         STRAS TYPE STRAS,  
         ADRNR TYPE ADRNR,  
         MATNR TYPE MATNR,  
         MAKTX TYPE MAKTX,  
         MAKTG TYPE MAKTG,  
       END OF TY\_PO.  
  
DATA: LT\_PO TYPE TABLE OF TY\_PO,  
      LS\_PO TYPE TY\_PO.  
  
DATA: LV\_EBELN TYPE EBELN.  
SELECT-OPTIONS S\_EBELN FOR LV\_EBELN.

START-OF-SELECTION.  
  
 SELECT EKKO~EBELN,  
         EKKO~BUKRS,  
         EKPO~EBELP,  
         EKPO~EMATN,  
         EKPO~LGORT,  
         EKPO~MATKL,  
         LFA1~LIFNR,  
         LFA1~NAME1,  
         LFA1~ORT01,  
         LFA1~ORT02,  
         LFA1~PFACH,  
         LFA1~PSTLZ,  
         LFA1~STRAS,  
         LFA1~ADRNR,  
         MAKT~MATNR,  
         MAKT~MAKTX,  
         MAKT~MAKTG INTO TABLE @LT\_PO FROM EKKO AS EKKO  
  INNER JOIN EKPO AS EKPO ON EKKO~EBELN EQ EKPO~EBELN  
  INNER JOIN LFA1 AS LFA1 ON EKKO~LIFNR EQ LFA1~LIFNR  
  INNER JOIN MAKT AS MAKT ON EKPO~MATNR EQ MAKT~MATNR WHERE EKPO~EBELN IN @S\_EBELN.

 WRITE:/4 'EBELN' COLOR 3,12 'BUKRS' COLOR 3, 'EBELP' COLOR 3,26 'EMATN' COLOR 3,35 'LGORT' COLOR 3,  
 43 'MATKL' COLOR 3,55 'LIFNR' COLOR 3,65 'NAME1' COLOR 3,70 'ORT01' COLOR 3,80 'ORT02' COLOR 3,  
 91  'PFACH' COLOR 3, 'PSTLZ' COLOR 3 , 'STARS' COLOR 3, 'ADRNR' COLOR 3, 'MATNR' COLOR 3,136 'MATKX' COLOR 3, 'MATKG' COLOR 3.  
  LOOP AT LT\_PO INTO LS\_PO.  
    WRITE: / LS\_PO-EBELN,  
            LS\_PO-BUKRS,  
          18 LS\_PO-EBELP,  
            LS\_PO-EMATN,  
            LS\_PO-LGORT,  
            LS\_PO-MATKL,  
            LS\_PO-LIFNR,  
            LS\_PO-NAME1,  
            LS\_PO-ORT01,  
            LS\_PO-ORT02,  
            LS\_PO-PFACH,  
            LS\_PO-PSTLZ,  
            LS\_PO-STRAS,  
            LS\_PO-ADRNR,  
            LS\_PO-MATNR,  
            LS\_PO-MAKTX,  
            LS\_PO-MAKTG.  
  ENDLOOP.

Output



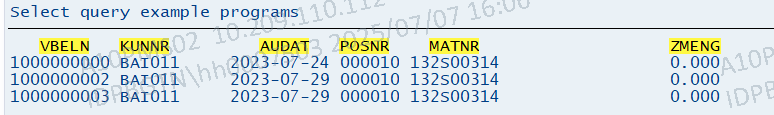
1.7.2 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_PO,  
         EBELN TYPE EBELN,  
         BUKRS TYPE BUKRS,  
         EBELP TYPE EBELP,  
         EMATN TYPE EMATN,  
         LGORT TYPE EKPO-LGORT,  
         MATKL TYPE MATKL,  
       END OF TY\_PO.  
  
DATA: LT\_PO TYPE TABLE OF TY\_PO,  
      LS\_PO TYPE TY\_PO.  
  
START-OF-SELECTION.  
  
  SELECT A~EBELN,  
         A~BUKRS,  
         B~EBELP,  
         B~EMATN,  
         B~LGORT,  
         B~MATKL INTO TABLE @LT\_PO FROM EKKO AS A  
  LEFT JOIN EKPO AS B ON A~EBELN EQ B~EBELN.  
  
  WRITE:/4 'EBELN' COLOR 3,12 'BUKRS' COLOR 3, 'EBELP' COLOR 3,26 'EMATN' COLOR 3,70 'LGORT' COLOR 3,  
 65 'MATKL' COLOR 3.  
  LOOP AT LT\_PO INTO LS\_PO.  
    WRITE: / LS\_PO-EBELN,  
            LS\_PO-BUKRS,  
          18 LS\_PO-EBELP,  
            LS\_PO-EMATN,  
            LS\_PO-LGORT,  
            LS\_PO-MATKL.  
  ENDLOOP.

Output



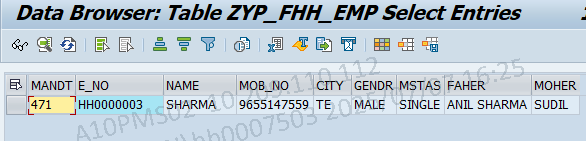
1.8 REPORT ZYP\_SELECT\_QUERY.  
  
TYPES: BEGIN OF TY\_VBAK,  
         VBELN LIKE VBAK-VBELN,  
         KUNNR LIKE VBAK-KUNNR,  
         AUDAT LIKE VBAK-AUDAT,  
       END OF TY\_VBAK.  
  
TYPES: BEGIN OF TY\_VBAP,  
         VBELN LIKE VBAP-VBELN,  
         POSNR LIKE VBAP-POSNR,  
         MATNR LIKE VBAP-MATNR,  
         ZMENG LIKE VBAP-ZMENG,  
       END OF TY\_VBAP.  
  
TYPES: BEGIN OF TY\_SO,  
         VBELN LIKE VBAK-VBELN,  
         KUNNR LIKE VBAK-KUNNR,  
         AUDAT LIKE VBAK-AUDAT,  
         POSNR LIKE VBAP-POSNR,  
         MATNR LIKE VBAP-MATNR,  
         ZMENG LIKE VBAP-ZMENG,  
       END OF TY\_SO.  
  
DATA: LT\_VBAK  TYPE TABLE OF TY\_VBAK,  
      LS\_VBAK  TYPE TY\_VBAK,  
      LT\_VBAP  TYPE TABLE OF TY\_VBAP,  
      LS\_VBAP  TYPE TY\_VBAP,  
      LT\_SO    TYPE TABLE OF TY\_SO,  
      LS\_SO    TYPE TY\_SO,  
      LV\_INDEX TYPE SY-TABIX.  
  
SELECT VBELN KUNNR AUDAT  
INTO TABLE LT\_VBAK  
FROM VBAK.  
  
CHECK SY-SUBRC = 0.  
  
SELECT VBELN POSNR MATNR ZMENG  
INTO TABLE LT\_VBAP  
FROM VBAP  
FOR ALL ENTRIES IN LT\_VBAK  
WHERE VBELN = LT\_VBAK-VBELN.  
  
CHECK SY-SUBRC = 0.  
  
SORT LT\_VBAP BY VBELN POSNR.  
  
LOOP AT LT\_VBAK INTO LS\_VBAK.  
  READ TABLE LT\_VBAP INTO LS\_VBAP WITH KEY VBELN = LS\_VBAK-VBELN BINARY SEARCH.  
  IF SY-SUBRC = 0.  
    MOVE-CORRESPONDING LS\_VBAP TO LS\_SO.  
    MOVE-CORRESPONDING LS\_VBAK TO LS\_SO.  
    APPEND LS\_SO TO LT\_SO.  
  ENDIF.  
ENDLOOP.  
  
WRITE: 'VBELN', 'KUNNR', 'AUDAT', 'POSNR', 'MATNR', 'ZMENG'.  
LOOP AT LT\_SO INTO LS\_SO.  
  WRITE: / LS\_SO-VBELN,  
           LS\_SO-KUNNR,  
           LS\_SO-AUDAT,  
           LS\_SO-POSNR,  
           LS\_SO-MATNR,  
           LS\_SO-ZMENG.  
ENDLOOP.

Output

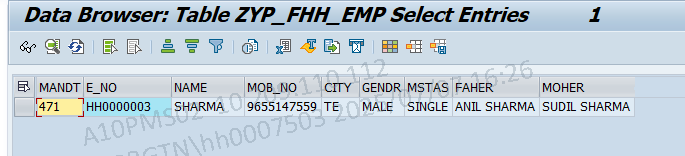


1.9.1 REPORT ZYP\_SELECT\_QUERY.  
  
PARAMETERS: P\_E\_NO   TYPE ZYP\_EMP\_NO,  
            P\_NAME   TYPE ZYP\_NAME,  
            P\_MOB\_NO TYPE ZYP\_MOB\_NO,  
            P\_CITY   TYPE ZYP\_PLACE,  
            P\_GENDR  TYPE C LENGTH 6,  
            P\_MSTAS  TYPE C LENGTH 8,  
            P\_FAHER  TYPE C LENGTH 25,  
            P\_MOHER  TYPE C LENGTH 25.  
  
DATA: LT\_EMP\_DET TYPE TABLE OF ZYP\_FHH\_EMP,  
      LS\_EMP\_DET LIKE LINE OF LT\_EMP\_DET.  
  
LS\_EMP\_DET-E\_NO     =    P\_E\_NO.  
LS\_EMP\_DET-NAME     =    P\_NAME.  
LS\_EMP\_DET-MOB\_NO   =    P\_MOB\_NO.  
LS\_EMP\_DET-CITY     =    P\_CITY.  
LS\_EMP\_DET-GENDR    =    P\_GENDR.  
LS\_EMP\_DET-MSTAS    =    P\_MSTAS.  
LS\_EMP\_DET-FAHER    =    P\_FAHER.  
LS\_EMP\_DET-MOHER    =    P\_MOHER.  
APPEND LS\_EMP\_DET TO LT\_EMP\_DET.  
CLEAR: LS\_EMP\_DET.  
  
MODIFY ZYP\_FHH\_EMP FROM TABLE LT\_EMP\_DET.

Output create

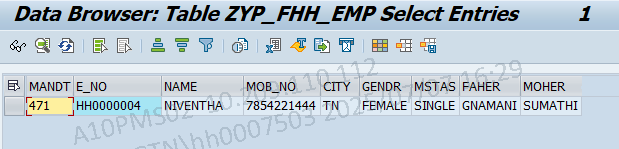


Output Modify



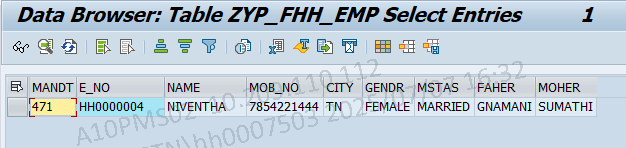
1.9.2 REPORT ZYP\_SELECT\_QUERY.  
  
PARAMETERS: P\_E\_NO   TYPE ZYP\_EMP\_NO,  
            P\_NAME   TYPE ZYP\_NAME,  
            P\_MOB\_NO TYPE ZYP\_MOB\_NO,  
            P\_CITY   TYPE ZYP\_PLACE,  
            P\_GENDR  TYPE C LENGTH 6,  
            P\_MSTAS  TYPE C LENGTH 8,  
            P\_FAHER  TYPE C LENGTH 25,  
            P\_MOHER  TYPE C LENGTH 25.  
  
DATA: LT\_EMP\_DET TYPE TABLE OF ZYP\_FHH\_EMP,  
      LS\_EMP\_DET LIKE LINE OF LT\_EMP\_DET.  
  
LS\_EMP\_DET-E\_NO     =    P\_E\_NO.  
LS\_EMP\_DET-NAME     =    P\_NAME.  
LS\_EMP\_DET-MOB\_NO   =    P\_MOB\_NO.  
LS\_EMP\_DET-CITY     =    P\_CITY.  
LS\_EMP\_DET-GENDR    =    P\_GENDR.  
LS\_EMP\_DET-MSTAS    =    P\_MSTAS.  
LS\_EMP\_DET-FAHER    =    P\_FAHER.  
LS\_EMP\_DET-MOHER    =    P\_MOHER.  
APPEND LS\_EMP\_DET TO LT\_EMP\_DET.  
CLEAR: LS\_EMP\_DET.  
  
INSERT ZYP\_FHH\_EMP FROM TABLE LT\_EMP\_DET.

Output:



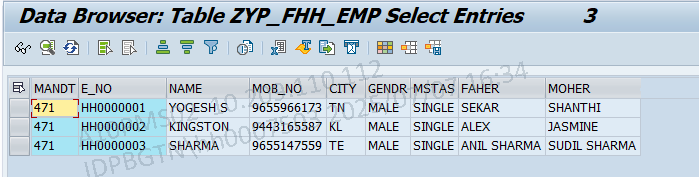
1.9.3 REPORT ZYP\_SELECT\_QUERY.  
  
PARAMETERS: P\_E\_NO   TYPE ZYP\_EMP\_NO,  
            P\_NAME   TYPE ZYP\_NAME,  
            P\_MOB\_NO TYPE ZYP\_MOB\_NO,  
            P\_CITY   TYPE ZYP\_PLACE,  
            P\_GENDR  TYPE C LENGTH 6,  
            P\_MSTAS  TYPE C LENGTH 8,  
            P\_FAHER  TYPE C LENGTH 25,  
            P\_MOHER  TYPE C LENGTH 25.  
  
DATA: LT\_EMP\_DET TYPE TABLE OF ZYP\_FHH\_EMP,  
      LS\_EMP\_DET LIKE LINE OF LT\_EMP\_DET.  
  
LS\_EMP\_DET-E\_NO     =    P\_E\_NO.  
LS\_EMP\_DET-NAME     =    P\_NAME.  
LS\_EMP\_DET-MOB\_NO   =    P\_MOB\_NO.  
LS\_EMP\_DET-CITY     =    P\_CITY.  
LS\_EMP\_DET-GENDR    =    P\_GENDR.  
LS\_EMP\_DET-MSTAS    =    P\_MSTAS.  
LS\_EMP\_DET-FAHER    =    P\_FAHER.  
LS\_EMP\_DET-MOHER    =    P\_MOHER.  
APPEND LS\_EMP\_DET TO LT\_EMP\_DET.  
CLEAR: LS\_EMP\_DET.  
  
UPDATE ZYP\_FHH\_EMP FROM TABLE LT\_EMP\_DET.

Output:



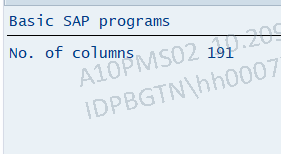
1.9.4 REPORT ZYP\_SELECT\_QUERY.  
  
PARAMETERS: P\_E\_NO   TYPE ZYP\_EMP\_NO,  
            P\_NAME   TYPE ZYP\_NAME,  
            P\_MOB\_NO TYPE ZYP\_MOB\_NO,  
            P\_CITY   TYPE ZYP\_PLACE,  
            P\_GENDR  TYPE C LENGTH 6,  
            P\_MSTAS  TYPE C LENGTH 8,  
            P\_FAHER  TYPE C LENGTH 25,  
            P\_MOHER  TYPE C LENGTH 25.  
  
DATA: LT\_EMP\_DET TYPE TABLE OF ZYP\_FHH\_EMP,  
      LS\_EMP\_DET LIKE LINE OF LT\_EMP\_DET.  
  
LS\_EMP\_DET-E\_NO     =    P\_E\_NO.  
LS\_EMP\_DET-NAME     =    P\_NAME.  
LS\_EMP\_DET-MOB\_NO   =    P\_MOB\_NO.  
LS\_EMP\_DET-CITY     =    P\_CITY.  
LS\_EMP\_DET-GENDR    =    P\_GENDR.  
LS\_EMP\_DET-MSTAS    =    P\_MSTAS.  
LS\_EMP\_DET-FAHER    =    P\_FAHER.  
LS\_EMP\_DET-MOHER    =    P\_MOHER.  
APPEND LS\_EMP\_DET TO LT\_EMP\_DET.  
CLEAR: LS\_EMP\_DET.  
  
DELETE ZYP\_FHH\_EMP FROM TABLE LT\_EMP\_DET.

Output:

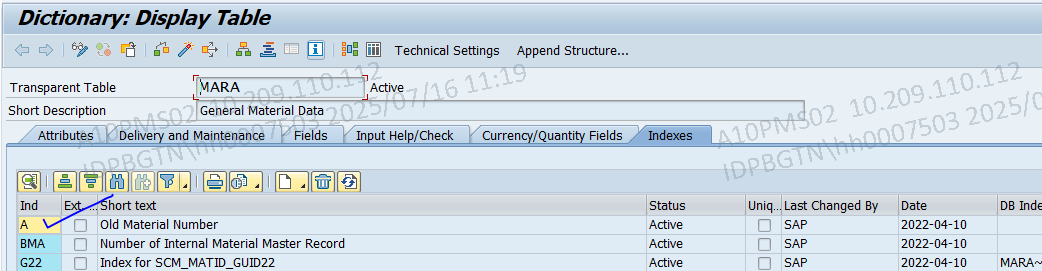


1.10 . DATA: LT\_MARA       TYPE TABLE OF EKKO,  
       LO\_TABLE\_DES    TYPE REF TO CL\_ABAP\_TABLEDESCR,  
       LO\_STRUCT\_DESCR TYPE REF TO CL\_ABAP\_STRUCTDESCR.  
  
TRY .  
    LO\_TABLE\_DES      ?= CL\_ABAP\_TABLEDESCR=>DESCRIBE\_BY\_DATA( P\_DATA = LT\_MARA ).  
    LO\_STRUCT\_DESCR   ?= LO\_TABLE\_DES->GET\_TABLE\_LINE\_TYPE( ).  
    DATA(LV\_NO\_OF\_COL) = LINES( LO\_STRUCT\_DESCR->COMPONENTS ).  
    WRITE: 'NO. OF COLUMNS', LV\_NO\_OF\_COL.  
  CATCH CX\_SY\_MOVE\_CAST\_ERROR.  
ENDTRY.

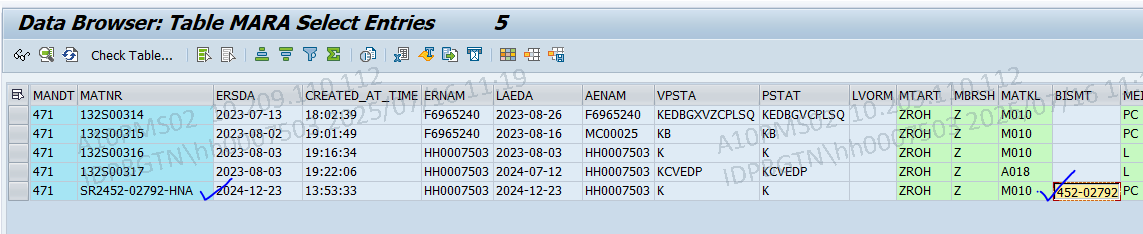
Output:



1.11 Secondary index example in material master table..



Data:



Source code:

REPORT ZYP\_DB.  
  
PARAMETERS P\_BISMT TYPE BISMT.  
  
DATA: LV\_MATNR TYPE MATNR,  
      LV\_BISMT TYPE BISMT.  
  
SELECT MATNR BISMT FROM MARA INTO (LV\_MATNR, LV\_BISMT)

WHERE BISMT = P\_BISMT %\_HINTS ORACLE 'INDEX ("MARA", "MARA~A")'.  
  WRITE:/  LV\_MATNR, LV\_BISMT.  
ENDSELECT.

Output:

