```
      ************ IDENTIFICATION DIVISION  ***************************
       IDENTIFICATION DIVISION.
       PROGRAM-ID. PROJECT2A as "PROJECT2A".
       DATE-WRITTEN. OCT-23-2016.
       DATE-COMPILED. OCT-24-2016.
SECURITY. PUBLIC.


      ************ ENVIRONMENT DIVISION  ******************************
       ENVIRONMENT DIVISION.
       CONFIGURATION SECTION.
       INPUT-OUTPUT SECTION.
      *    INVENT-FILE-IN IS THE INPUT PROVIDED BY THE CLIENT. THIS
      *    FILE WILL BE READ AND PROESSED, AND TWO REPORTS WILL BE
      *    MADE AS PER REQUIREMENTS:
      *    INVENT-REPORT-OUT:
      *        HAS INFORMATION ON PART NUMBER, PART NAME, QNTY ON HAND,
      *        UNIT PRICE.
      *    REORDER-REPORT-OUT:
      *        HAS INFORMATION ON PART NUMBER, QNTY, AND RE-ORDER POINT.
      *
       FILE-CONTROL.
           SELECT INVENT-FILE-IN ASSIGN TO "INVEN.TXT"
               FILE STATUS IS FILE-CHECK-KEY
               ORGANIZATION IS LINE SEQUENTIAL.

           SELECT INVENT-REPORT-OUT ASSIGN TO "INVREPRT.TXT"
               ORGANIZATION IS LINE SEQUENTIAL.

           SELECT REORDER-REPORT-OUT ASSIGN TO "REORDER-RPRT.TXT"
               ORGANIZATION IS LINE SEQUENTIAL.


      ************ DATA DIVISION  *************************************
       DATA DIVISION.

      *    FILE SECTION DESCRIBES THE LAYOUT OF INPUT/OUTPUT

       FILE SECTION.
      *    INVENT-FILE-IN:
      *        DESCRIBES THE RECORD LAYOUT OF THE INPUT:
      *        ONE RECORD HAS THE FOLLOWING FIELDS:
      *            PARTNUMBER
      *            PARTNAME
      *            QUANTITY ON HAND
      *            UNIT PRICE
      *            REORDER POINT
      *
       FD INVENT-FILE-IN.
       01 INPUT-RECORD.
           88 EOF              VALUE HIGH-VALUES.
           05 IN-PARTNUMBER    PIC 9(5).
           05 IN-PARTNAME      PIC X(20).
           05 IN-QNTY-ONHAND   PIC 9(3).
           05 IN-UNITPRICE     PIC 9(4).
           05 IN-REORER-POINT  PIC 9(3).
```

```
*    INVENT-REPORT-OUT:
*        DESCRIBES THE RECORD LAYOUT OF THE OUTPUT:
*        ONE RECORD OF THE INVENT-REPORT-OUT HAS THE FILEDS:
*            PRINTLINE WHICH IS A BLANK LINE OF MAXIMUM 41 BYTES.
*
*        THE BLANK LINE ACTS AS A PLACEHOLDER FOR ALL THE VALUES
*        THAT ARE HELD BY:
*
*            INVENT-HEADING-TITLE: TITLE OF THE REPORT. SINCE
*                THERE ARE TWO REPORTS GENERATED, IT WOULD BE
*                A GOOD IDEAD TO PUT A TITLE FOR THEM INCDICATING
*                THEIR PURPOSE.
*                THIS LINE IS WRITTEN ONLY ONCE.
*
*            INVENT-HEADING-LINE: COLUMN LABELS
*                THIS LINE IS WRITTEN ONLY ONCE.
*
*            DET-LINE: PLACEHOLDER FOR A SINGLE RECORD OF FIELDS.
*                THIS LINE IS ITERATIVELY WRITTEN INTO THE OUTPUT.
*
*            TOTAL-LINE: AS PART OF THE SUMMARY OF THE REPORT.
*                TO INDICATE THE ACCUMULATED VALUE OF UNIT PRICES.
*                THIS LINE IS WRITTEN ONLY ONCE.
*
*            REC-READ-LINE: AS PART OF THE SUMMARY OF THE REPORT.
*                TO INDICATE THE NUMBER OF RECORDS THAT WERE READ
*                FROM THE INPUT FILE.
*                THIS LINE IS WRITTEN ONLY ONCE.
*
*            REC-WRITTEN-LINE:AS PART OF THE SUMMARY OF THE
*                REPORT. TO INDICATE THE NUMBER OF RECORDS WRITTEN
*                INTO THE OUTPUT.
*                THIS LINE IS WRITTEN ONLY ONCE.
*
 FD INVENT-REPORT-OUT.
 01 OUTPUT-RECORD.
     05 PRINTLINE        PIC X(41).

 FD REORDER-REPORT-OUT.
 01 REORDER-OUTPUT-RECORD.
     05 PRINTLINE        PIC X(41).

 WORKING-STORAGE SECTION.
 01 INVENT-HEADING-TITLE.
     05 FILLER           PIC X(1) VALUE SPACES.
     05 FILLER           PIC X(17) VALUE 'INVENTORY REPORT'.

 01 INVENT-HEADING-LINE.
*    LEFT OFFSET
     05 FILLER           PIC X(1) VALUE SPACES.

*    REPORT COLUMN LABELS
     05 FILLER           PIC X(6) VALUE 'NUMBER'.
     05 FILLER           PIC X(1) VALUE SPACES.

     05 FILLER           PIC X(9) VALUE 'PART NAME'.
     05 FILLER           PIC X(13) VALUE SPACES.
```

```
           05 FILLER            PIC X(3) VALUE 'QTY'.
           05 FILLER            PIC X(2) VALUE SPACES.

           05 FILLER            PIC X(5) VALUE 'VALUE'.
           05 FILLER            PIC X(1) VALUE SPACES.

       01 REORDER-HEADING-TITLE.
           05 FILLER            PIC X(1) VALUE SPACES.
           05 FILLER            PIC X(15) VALUE 'REORDER REPORT'.

       01 REORDER-HEADING-LINE.
      *    LEFT OFFSET
           05 FILLER            PIC X(1) VALUE SPACES.

      *    REPORT COLUMN LABELS
           05 FILLER            PIC X(6) VALUE 'NUMBER'.
           05 FILLER            PIC X(1) VALUE SPACES.

           05 FILLER            PIC X(3) VALUE 'QTY'.
           05 FILLER            PIC X(4) VALUE SPACES.

           05 FILLER            PIC X(14) VALUE 'RE-ORDER POINT'.
           05 FILLER            PIC X(1) VALUE SPACES.

       01 DET-LINE.
      *    LEFT OFFSET
           05 FILLER            PIC X(1) VALUE SPACES.

      *    REPORT VALUES
           05 DET-PARTNUMBER    PIC 9(5).
           05 FILLER            PIC X(2) VALUE SPACES.

           05 DET-PARTNAME      PIC X(20).
           05 FILLER            PIC X(2) VALUE SPACES.

           05 DISP-QTY-ON-HAND  PIC Z(2)9.
           05 FILLER            PIC X(2) VALUE SPACES.

           05 FILLER            PIC X(2) VALUE SPACES.
           05 DISP-VALUE        PIC Z(5)9.

       01 REORDER-DET-LINE.
      *    LEFT OFFSET
           05 FILLER            PIC X(1) VALUE SPACES.

      *    REPORT VALUES
           05 RE-PARTNUMBER     PIC 9(5).
           05 FILLER            PIC X(2) VALUE SPACES.

           05 RE-QTY-ON-HAND    PIC 9(3).
           05 FILLER            PIC X(2) VALUE SPACES.

           05 FILLER            PIC X(2) VALUE SPACES.
           05 REORDER-PNT       PIC 9(3).

       01 TOTAL-LINE.
```

```
*    LEFT OFFSET
     05 FILLER            PIC X(8) VALUE SPACES.

     05 FILLER            PIC X(11) VALUE 'TOTAL VALUE'.
     05 FILLER            PIC X(3) VALUE SPACES.
     05 DISP-TOTAL-VALUE  PIC Z(9)9.

 01 REC-READ-LINE.
*    LEFT OFFSET
     05 FILLER            PIC X(8) VALUE SPACES.

     05 FILLER            PIC X(12) VALUE 'RECORDS READ'.
     05 FILLER            PIC X(4) VALUE SPACES.
     05 DIS-READ-COUNTER  PIC Z(4)9.

 01 REC-WRITTEN-LINE.
*    LEFT OFFSET
     05 FILLER            PIC X(8) VALUE SPACES.

     05 FILLER            PIC X(15) VALUE 'RECORDS WRITTEN'.
     05 FILLER            PIC X(1) VALUE SPACES.
     05 DIS-WRITE-COUNTER PIC Z(4)9.

****************************************************************
*    VARIABLES TO HOLD VALUES, TEMPORARILY BEFORE PASSING THEM
*    TO THEIR PRE-FIXED DIS- or DISP- PREFEIXED VARIABLES FOR
*    BETETR READABILITY.
*    i.e. INSTEAD OF SHOWING TOTAL VALUE AS 00003000, WE
*    SHOW IT AS 3000.
*
*    TOTAL-VALUE: ACCUMULATED VALUES OF UNIT PRICES.
*    UNITPRICE: THE UNIT PRICE.
*    QNTY-ONHAND: QUANTITY ON HAND FOR EACH ITEM.
*    WRITE-COUNTER: TO HOLD THE NUMBER OF ENTRIES WRITTEN INTO
*        THE REPORT.
*    READ-COUNTER: TO HOLD THE NUMBER OF ENTRIES READ SUCCESSFULY
*        FROM THE INPUT FILE.
*    FILE-CHECK-KEY HOLDS THE RETURNING VALUE AT THE POINT WHERE
*    THE PROGRAM TRIES TO OPEN THE INPUT FILE:
*    CODE 00: SUCCESSFUL COMPLETION.
*        FOR MORE INFOTMATION ON THE CODES, PLEASE REFER TO:
*        GOOGLE "FILE STATUS CODE TABLES - MICRO FOCUS"
*

 01 TOTAL-VALUE          PIC 9(10) VALUE ZEROS.
 01 UNITPRICE            PIC 9(4).
 01 QNTY-ONHAND          PIC 9(3).
 01 WRITE-COUNTER        PIC 9(5) VALUE ZERO.
 01 READ-COUNTER         PIC 9(5) VALUE ZERO.
 01 FILE-CHECK-KEY       PIC X(2).

************ PROCEDURE DIVISION  ******************************
 PROCEDURE DIVISION.
*    0000-MAINLINE:
*    IS A LEVEL 0 PROGRAM, ON TOP, CONTROLLING ALL SUBROUTINES.
*    MAINLINE IS THE MAIN PROCESS.
*    IT HAS 3 SUB-ROUTINES:
```

```
*        INITIATE: TO OPEN THE FILES, AND READ THE FIRST RECORD
*                  IN THE INPUT FILE (IF IT EXISTS).
*        PROCESS:  TO WRITE THE HEADINGS OF THE REPORTS.
*                  TO WRITE THE DETAILS OF THE REPORTS.
*                  TO WRITE THE SUMMARY OF THE INVENRPRT.TXT
*        CLOSE:    TO CLOSE ALL FILES, INPUTS AND OUTPUTS.
*                  TO TO TEMINATE THE PROGRAM.
*
 0000-MAINLINE.
     PERFORM 0100-INITIALIZE.
     PERFORM 0100-PROCESS.
     PERFORM 0100-CLOSE.

********** SUBROUTINE ****************************************
*    0100-INITIALIZE;
*    IS A LEVEL 1 SUBROUTINE.
*        OPEN FILES: OPEN ALL INPUT AND OUTPUT FILES.
*        READ-RECORD: PRIME-READ THE INPUT FILE; THE FIRST RECORD
*            IS READ, IF IT EXISTS OR IF THE FILE ITSELF EXISTS.

 0100-INITIALIZE.
     PERFORM 0200-OPEN-FILES.
     PERFORM 0200-READ-RECORD.

********** SUBROUTINE ****************************************
*    0100-PROCESS:
*    IS LEVEL 1 SUBROUTINE.
*        WRITE-HEADING: WRITE THE HEADING FOR OUTPUTS.
*        WRITE-DETAIL-REPORT: ITERATIVELY WRITE ONE RECORD AT A
*            TIME UNTIL THE END OF FILE (EOF) FLAG IS SET SET TO
*            TRUE. IF SO, RETURN CONTROL AND PROCEED TO
*        WRITE-SUMMARY: WRTIE THE SUMMARY AT THE BOTTOM OF THE
*            REPORT.
 0100-PROCESS.
     PERFORM 0200-WRITE-HEADING.
     PERFORM 0200-WRITE-DETAIL-REPORT UNTIL EOF.
     PERFORM 0200-WRITE-SUMMARY.

********** SUBROUTINE ****************************************
*    0100-CLOSE:
*    IS A LEVEL 1 SUBROUTINE.
*        CLOSE FILES: CLOSE ALL FILES.
*        TERMINATE THE PROGRAM.
 0100-CLOSE.
     PERFORM 0200-CLOSE-FILES.
     PERFORM 0200-CLOSE-PROGRAM.

********** SUBROUTINE ****************************************
*    0200-OPEN-FILES:
*    IS A LEVEL 2 SUBROUTINE.
*        IT OPENS THE INPUT FILE: CHECKS TO SEE IF THE FILE CAN
*            SUCCESSFULY BE OPENED. IF NOT, AN ERROR MSG IS
*            DISPLAYED AND THE PROGRAM GOES IN FORWARD DIRECTION
*            TO TERMINATION OF THE PROGRAM (i.e. IF THERE IS NO
*            INPUT, THERE WILL NEVER BE A POINT OF USING
*            RESOURCES TO PRODUCE AN OUTPUT.
*        IT OPENS THE OUTPUT FILES: BOTH OUTPUT FILES ARE OPENED
```

```
*            TO BE WRITTEN.
*
 0200-OPEN-FILES.
     OPEN INPUT INVENT-FILE-IN.
     IF FILE-CHECK-KEY NOT = "00"
         THEN DISPLAY "THERE WAS A PROBELM OPENING THE INPUT FILE"
         GO TO 0200-CLOSE-PROGRAM
     END-IF.
     OPEN OUTPUT INVENT-REPORT-OUT.
     OPEN OUTPUT REORDER-REPORT-OUT.

********** SUBROUTINE ****************************************
*    0200-READ-RECORD:
*    IS A LEVEL 2 SUBROUTINE.
*        IT READS A RECORD.
*        IF THE POINTER HAS REACHED THE END OF FILE, WE SET THE
*        END OF FILE (EOF) FLAG TO TRUE, IF NOT, WE ADD ONE AND
*        UPDATE OUT READ-COUNTER.
 0200-READ-RECORD.
     READ INVENT-FILE-IN
         AT END SET EOF TO TRUE
         NOT AT END ADD 1 TO READ-COUNTER
     END-READ.

********** SUBROUTINE ****************************************
*    0200-WRITE-HEADING:
*    IS A LEVEL 2 SUBROUTINE.
*    WRITES THE TITLE FOR INVENT-REPORT-OUT, ONCE, AND PLACES
*        A BLANK LINE AFTER THAT.
*    WRITES THE COLUMN LABELS FOR INVENT-REPORT-OUT, ONCE, AFTER
*        PLACING 3 BLANK LINES.
*    WRITES THE TITLE FOR REORDER-OUTPUT-RECORD, ONCE, AND PLACES
*        A BLANK LINE AFTER THAT.
*    WRITES THE COLUMN LABELS FOR REORDER-OUTPUT-RECORD, ONCE,
*        AFTER PLACING 3 BLANK LINES.
*
 0200-WRITE-HEADING.
     WRITE OUTPUT-RECORD FROM INVENT-HEADING-TITLE
         BEFORE ADVANCING 1 LINE.
     WRITE OUTPUT-RECORD FROM INVENT-HEADING-LINE
         AFTER ADVANCING 3 LINES.
     WRITE REORDER-OUTPUT-RECORD FROM REORDER-HEADING-TITLE
         BEFORE ADVANCING 1 LINE.
     WRITE REORDER-OUTPUT-RECORD FROM REORDER-HEADING-LINE
         AFTER ADVANCING 3 LINES.

********** SUBROUTINE ****************************************
*    0200-WRITE-DETAIL-REPORT:
*    IS A LEVEL 2 SUBROUTINE.
*    IS PERFORMED ITERATIVELY UNTIL THE END OF FILE FLAG IS SET
*        TRUE:
*        WRITE ONE RECORD AT A TIME
*        READ ONE RECORD.
*
 0200-WRITE-DETAIL-REPORT.
     PERFORM 0300-WRITE-ONE-RECORD.
     PERFORM 0200-READ-RECORD.
```

```
********** SUBROUTINE ****************************************
*    0200-WRITE-SUMMARY:
*    IS A LEVEL 2 SUBROUTINE.
*    IT COPIES VALUES TO THEIR RESPECTIVE DIS- PREFIXED PLACEHOLDE
*        -RS FOR BETTER READABILITY.
*    WRITES TOTAL VALUE SUMMARY AT THE BOTTOM OF THE REPORT AFTER
*        PLACING 4 BLANK LINES, AS PER LAYOUT REQUIREMENTS.
*    WRITES THE NUMBER OF RECORDS READ FROM THE INPUT FILE.
*    WRITES THE NUMBER OF RECORDS WRITTEN TO THE OUTPUT FILE.
*
 0200-WRITE-SUMMARY.
     MOVE READ-COUNTER TO DIS-READ-COUNTER.
     MOVE WRITE-COUNTER TO DIS-WRITE-COUNTER.
     WRITE OUTPUT-RECORD FROM TOTAL-LINE AFTER ADVANCING 4 LINES.
     WRITE OUTPUT-RECORD FROM REC-READ-LINE
         AFTER ADVANCING 1 LINE.
     WRITE OUTPUT-RECORD FROM REC-WRITTEN-LINE
         AFTER ADVANCING 1 LINE.

********** SUBROUTINE ****************************************
*    0200-CLOSE-FILES:
*    IS A LEVEL 2 SUBROUTINE.
*    CLOSES ALL THE INPUT AND OUTPUT FILES.
*
 0200-CLOSE-FILES.
     CLOSE INVENT-FILE-IN, INVENT-REPORT-OUT, REORDER-REPORT-OUT.

********** SUBROUTINE ****************************************
*    0200-CLOSE-PROGRAM:
*    IS A LEVEL 2 SUBROUTINE.
*    WHEN CALLED, IT STOPS THE PROCESSING.
*
 0200-CLOSE-PROGRAM.
     STOP RUN.

********** SUBROUTINE ****************************************
*    0300-WRITE-ONE-RECORD:
*    IS A LEVEL 3 SUBROUTINE.
*    IT COPIES VALUES TO THEIR RESPECTIVE DIS- PREFIXED PLACEHOLDE
*        -RS FOR BETTER READABILITY.
*    COMPUTES AND UPDATES TOTAL VALUE BASED ON UNIT PRICE FOR
*        EACH ITEM IN THE INPUT.
*    ADDS 1 TO THE WRITE-COUNTER THAT HOLDS THE NUMBER OF
*        ENTRIES WRITTEN SUCCESSFULLY INTO THE OUTPUT.
*    CALLS THE 0400-WRITE-ONE-REORDER-RECORD SUBROUTINE IF:
*        THE QUANTITY ON HAND IS LESS THAN OR EQUAL TO
*        RE-OPRDER POINT OF THE CURRENT INPUT RECORD.
*
 0300-WRITE-ONE-RECORD.
     MOVE IN-PARTNAME TO DET-PARTNAME.
     MOVE IN-PARTNUMBER TO DET-PARTNUMBER.
     MOVE IN-QNTY-ONHAND TO QNTY-ONHAND.
     MOVE IN-UNITPRICE TO UNITPRICE.
     MOVE IN-REORER-POINT TO REORDER-PNT.
     MOVE IN-PARTNUMBER TO RE-PARTNUMBER.
     MOVE IN-QNTY-ONHAND TO RE-QTY-ON-HAND.
```

```
           COMPUTE TOTAL-VALUE = TOTAL-VALUE + (UNITPRICE).

           MOVE TOTAL-VALUE TO DISP-TOTAL-VALUE.
           MOVE UNITPRICE TO DISP-VALUE.
           MOVE QNTY-ONHAND TO DISP-QTY-ON-HAND.

           WRITE OUTPUT-RECORD FROM DET-LINE.
           ADD 1 TO WRITE-COUNTER.


           IF QNTY-ONHAND <= REORDER-PNT THEN
           PERFORM 0400-WRITE-ONE-REORDER-RECORD.

      ********** SUBROUTINE ****************************************
      *    0400-WRITE-ONE-REORDER-RECORD:
      *    IS A LEVEL 4 SUBROUTINE.
      *    IT IS CALLED ONLY IF THE CONDITION BELOW IS MET:
      *        IF THE QUANTITY ON HAND IS LESS THAN OR EQUAL TO
      *        RE-OPRDER POINT OF THE CURRENT INPUT RECORD.
      *    WRITES TO THE REORDER-OUTPUT-RECORD AFTER PLACING A SINGLE
      *        BLANK LINE.
      *
       0400-WRITE-ONE-REORDER-RECORD.
           WRITE REORDER-OUTPUT-RECORD FROM REORDER-DET-LINE
               AFTER ADVANCING 1 LINE.

      ********** END ************************************************
      *    INDICATES END OF THE PROGRAM.
      *
       END PROGRAM PROJECT2A.
```