

SORT COMMAND FORMAT

```

SORT sort-work-file-name
      / \
      |   |
ON < | ASCENDING | > KEY data-name-1 of
      | DESCENDING |      sort-work-file-rec
      |             |      [data-name-2]
      \             /
      .
      .

```

[WITH DUPLICATES IN ORDER]

[COLLATING SEQUENCE IS alphabet-name]

USING source-file-name-1
[source-file-name-2]

or

INPUT PROCEDURE IS procedure-name-1
(a paragraph name?)

GIVING sorted-output-file-name-1
[sorted-output-file-name-2] .
(braces here on both sides)

or

OUTOUT PROCEDURE IS procedure-name-2.
(a paragraph name?)

(may repeat ASCENDING and/or DESCENDING entries with the respective key(s))

EXAMPLE 1: *SAMPLE CODING WITH USING & GIVING CLAUSES*

ENVIRONMENT DIVISION.

.

INPUT-OUTPUT SECTIONS.

SELECT SALES-FILE
 ASSIGN TO "C:\SALES.99"
 ORGANIZATION IS LINE SEQUENTIAL.

SELECT SORT-WORK-FILE
 ASSIGN TO "C:\SRT-WRK.FIL".

SELECT SORTED-SALES-FILE
 ASSIGN TO "C:\SRTDSALS.99"
 ORGANIZATION IS LINE SEQUENTIAL.

DATA DIVISION.

FILE SECTION.

FD SALES-FILE.
01 INPUT-SALES-RECORD PIC X(39).

SD SORT-WORK-FILE.
01 SORT-RECORD.
 05 PIC X(4).
 05 S-SSN PIC 9(9).
 05 S-SALES-PERSON PIC X(20).
 05 PIC X(6).

FD SORTED-SALES-FILE.
01 SRTD-SALES-REC PIC X(39).

PROCEDURE DIVISION.

000-PRODUCE-SORTED-SALES-FILE.

SORT SORT-WORK-FILE
 ON ASCENDING S-SALES-PERSON
 S-SSN
 USING SALES-FILE
 GIVING SORTED-SALES-FILE.

STOP RUN.

```
RETURN sort-file-name RECORD [INTO identifier]
      AT END imperative-statement-1
      [NOT AT END imperative-statemtn-2]    (Don't use this clause!)
[END-RETURN]
```

EXAMPLE 2: *SAMPLE CODING WITH OUTPUT PROCEDURE CLAUSE.*

PROCEDURE DIVISION.

000-PRODUCE-SORTED-SALES-FILE.

```
    SORT SORT-WORK-FILE
      ON ASCENDING KEY    S-SALES-PERSON
                        S-SSN
    USING SALES-FILE
    OUTPUT PROCEDURE IS
      A000-PREPARE-REPORT.
```

STOP RUN.

A000-PREPARE-REPORT.

```
    PERFORM A100-START-UP.
    PERFORM A200-PROCESS-SORTED-RECORDS
      UNTIL NO-MORE-SORTED-RECORDS.
    PERFORM A300-CLOSURE-ROUTINES.
```

A100-START-UP.

```
    OPEN OUTPUT-SALES-REPORT.    (a print file; not defined above.)
      .
      .    (getdate, headers, etc.)
      .
    PERFORM X100-RETURN-SORT-RECORD.
```

A200-PROCESS-SORT-WORK-FILE.

```
    PERFORM A210-MOVE-AND-WRITE-DATA.
    PERFORM X100-RETURN-SORT-RECORD.
```

A210-MOVE-AND-WRITE-DATA.

```
    (move data to a detail line defined in WS)
    WRITE PRINT-REC FROM detail-line
      AFTER ADVANCING 1 LINE.
```

A300-CLOSURE-ROUTINE.

```
    CLOSE SALES-REPORT.
```

X100-RETURN-SORT-RECORD.

```
    RETURN SORT-FILE INTO SORT-REC-INPUT
      AT END
      SET WSnn-NO-MORE-SORT-RECORDS TO TRUE.
```

RELEASE sort-record-name [FROM identifier].

EXAMPLE 2: *SAMPLE CODING WITH OUTPUT PROCEDURE CLAUSE.*

(revised definition of the sort record)

```
01  SORT-RECORD.
    05                                     PIC X(4).
    05  S-SSN                             PIC 9(9).
    05  S-SALES-PERSON                     PIC X(20).
    05  S-CURRENT-SALES                     PIC 9(4)V99.
```

.

PROCEDURE DIVISION.

```
000-MAIN-CONTROL.
    SORT SORT-WORK-FILE
        DESCENDING  S-CURRENT-SALES
        ASCENDING   S-SALES-PERSON
                   S-SSN
    INPUT PROCEDURE
        A000-SELECT-SALES-RECORD
        GIVING SELECTED-SALES-FILE.
```

STOP RUN.

```
A000-SELECT-SALES-RECORD.
    PERFORM A100-START-UP.
    PERFORM A200-PROCESS-SALES-FILE
        UNTIL NO-MORE-RECORDS.
    PERFORM A300-CLOSURES.
```

```
A100-START-UP.
    OPEN INPUT SALES-FILE.
    .
    .
    .
    PERFORM X100-READ-SALES-FILE.
```

```
A200-PROCESS-SALES-FILE.
    IF CURRENT-SALES > 1200
        RELEASE SORT-RECORD FROM WSnn-INPUT-RECORD-AREA.
    PERFORM X100-READ-SALES-FILE.
```

```
A300-CLOSURES.
    CLOSE SALES-FILE.
```

```
X100-READ-SALES-FILE.
    READ SALES-FILE INTO WSnn-INPUT-RECORD-AREA
    AT END
```

SORTS

MIS 102

OVERHEAD NO. 5

SET WSNN-NO-MORE-SALES-RECORDS TO TRUE.