

COBOL - INTERNAL SORT

http://www.tutorialspoint.com/cobol/cobol_internal_sort.htm

Copyright © tutorialspoint.com

Sorting of data in a file or merging of two or more files is a common necessity in almost all business-oriented applications. Sorting is used for arranging records either in ascending or descending order, so that sequential processing can be performed. There are two techniques which are used for sorting files in COBOL:

- **External sort** is used to sort files by using the SORT utility in JCL. We have discussed this in the JCL chapter. As of now, we will focus on internal sort.
- **Internal sort** is used to sort files with in a COBOL program. **SORT** verb is used to sort a file.

Sort Verb

Three files are used in the Sort process in COBOL:

- **Input file** is the file which we have to sort either in ascending or descending order.
- **Work file** is used to hold records while the sort process is in progress. Input file records are transferred to the work file for the sorting process. This file should be defined in the File-Section under SD entry.
- **Output file** is the file which we get after sorting process. It is the final output of the Sort verb.

Syntax

Following is the syntax to sort a file:

```
SORT work-file ON ASCENDING KEY rec-key1  
                [ON DESCENDING KEY rec-key2]  
USING input-file GIVING output-file.
```

SORT performs the following operations:

- Opens work-file in the I-O mode, input-file in the INPUT mode and output-file in the OUTPUT mode.
- Transfers the records present in the input-file to the work-file.
- Sorts the SORT-FILE in ascending/descending sequence by rec-key.
- Transfers the sorted records from the work-file to the output-file.
- Closes the input-file and the output-file and deletes the work-file.

Example

In the following below example INPUT is the input file which needs to be sorted in ascending order:

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. HELLO.  
  
ENVIRONMENT DIVISION.  
INPUT-OUTPUT SECTION.  
FILE-CONTROL.  
    SELECT INPUT ASSIGN TO IN.  
    SELECT OUTPUT ASSIGN TO OUT.  
    SELECT WORK ASSIGN TO WRK.  
  
DATA DIVISION.  
FILE SECTION.
```

```

FD INPUT.
  01 INPUT-STUDENT.
    05 STUDENT-ID-I PIC 9(5).
    05 STUDENT-NAME-I PIC A(25).
FD OUTPUT.
  01 OUTPUT-STUDENT.
    05 STUDENT-ID-O PIC 9(5).
    05 STUDENT-NAME-O PIC A(25).
SD WORK.
  01 WORK-STUDENT.
    05 STUDENT-ID-W PIC 9(5).
    05 STUDENT-NAME-W PIC A(25).

PROCEDURE DIVISION.
  SORT WORK ON ASCENDING KEY STUDENT-ID-O
  USING INPUT GIVING OUTPUT.
  DISPLAY 'Sort Successful'.
STOP RUN.

```

JCL to execute the above COBOL program:

```

//SAMPLE JOB(TESTJCL,XXXXXX),CLASS=A,MSGCLASS=C
//STEP1 EXEC PGM=HELLO
//IN DD DSN=INPUT-FILE-NAME,DISP=SHR
//OUT DD DSN=OUTPUT-FILE-NAME,DISP=SHR
//WRK DD DSN=&&TEMP

```

When you compile and execute the above program, it produces the following result:

```
Sort Successful
```

Merge Verb

Two or more identically sequenced files are combined using Merge statement. Files used in the merge process:

- Input Files : Input-1, Input-2
- Work File
- Output File

Syntax

Following is the syntax to merge two or more files:

```

MERGE work-file ON ASCENDING KEY rec-key1
               [ON DESCENDING KEY rec-key2]

USING input-1, input-2 GIVING output-file.

```

Merge performs the following operations:

- Opens the work-file in the I-O mode, input-files in the INPUT mode and output-file in the OUTPUT mode.
- Transfers the records present in the input-files to the work-file.
- Sorts the SORT-FILE in ascending/descending sequence by rec-key.
- Transfers the sorted records from the work-file to the output-file.
- Closes the input-file and the output-file and deletes the work-file.

Example

In the following example, INPUT1 and INPUT2 are input the files which are to be merged in ascending order:

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. HELLO.  
  
ENVIRONMENT DIVISION.  
INPUT-OUTPUT SECTION.  
FILE-CONTROL.  
    SELECT INPUT1 ASSIGN TO IN1.  
    SELECT INPUT2 ASSIGN TO IN2.  
    SELECT OUTPUT ASSIGN TO OUT.  
    SELECT WORK ASSIGN TO WRK.  
  
DATA DIVISION.  
FILE SECTION.  
FD INPUT1.  
    01 INPUT1-STUDENT.  
        05 STUDENT-ID-I1 PIC 9(5).  
        05 STUDENT-NAME-I1 PIC A(25).  
FD INPUT2.  
    01 INPUT2-STUDENT.  
        05 STUDENT-ID-I2 PIC 9(5).  
        05 STUDENT-NAME-I2 PIC A(25).  
FD OUTPUT.  
    01 OUTPUT-STUDENT.  
        05 STUDENT-ID-O PIC 9(5).  
        05 STUDENT-NAME-O PIC A(25).  
SD WORK.  
    01 WORK-STUDENT.  
        05 STUDENT-ID-W PIC 9(5).  
        05 STUDENT-NAME-W PIC A(25).  
  
PROCEDURE DIVISION.  
    MERGE WORK ON ASCENDING KEY STUDENT-ID-O  
    USING INPUT1, INPUT2 GIVING OUTPUT.  
    DISPLAY 'Merge Successful'.  
STOP RUN.
```

JCL to execute the above COBOL program:

```
//SAMPLE JOB(TESTJCL,XXXXXX),CLASS=A,MSGCLASS=C  
//STEP1 EXEC PGM=HELLO  
//IN1 DD DSN=INPUT1-FILE-NAME,DISP=SHR  
//IN2 DD DSN=INPUT2-FILE-NAME,DISP=SHR  
//OUT DD DSN=OUTPUT-FILE-NAME,DISP=SHR  
//WRK DD DSN=&&TEMP
```

When you compile and execute the above program, it produces the following result:

```
Merge Successful
```