Job Control Language (JCL)

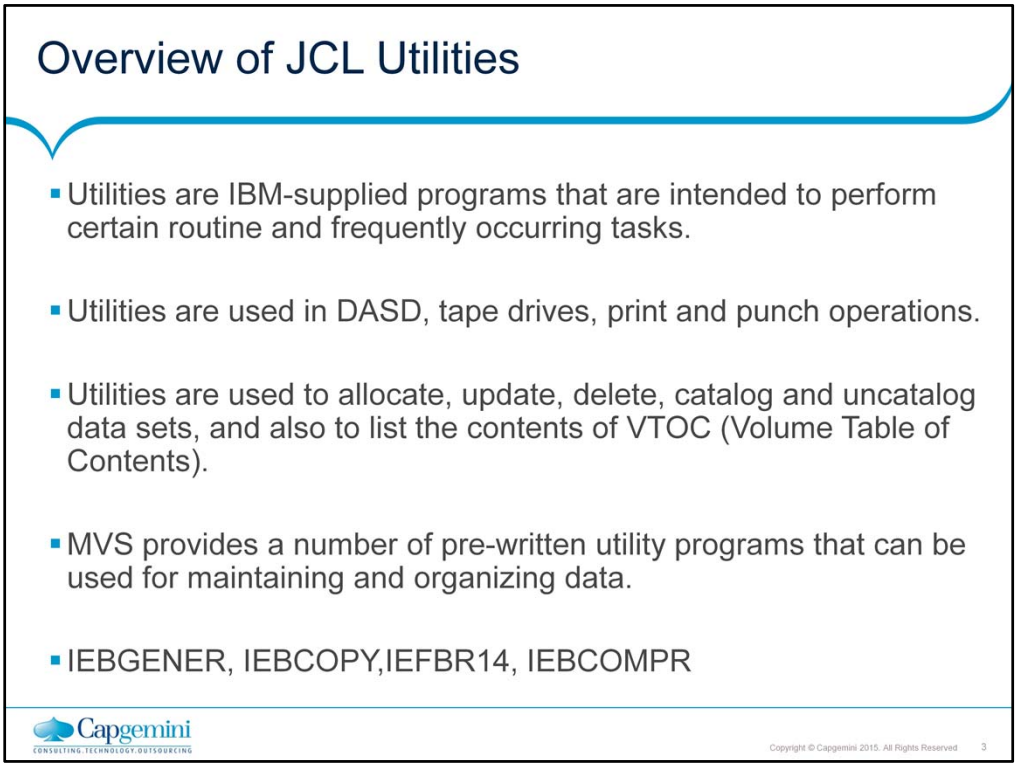
Lesson 7: Utility

Lesson Objectives

- IEFBR14 Utility
- IEBGENER Utility
- IEBCOPY
- IEHLIST
- SORT Utility

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 2



Utilities are broadly classified into two different categories:

Data set Utilities (prefixed with IEB)
System Utilities (prefixed with IEH)

Data set utilities are used to copy, print, update, reorganize, and compare data at the dataset and/or record level.

System utilities are used to list VTOC information, copy, delete, catalog and uncatalog datasets, to write tape labels and to add or delete dataset passwords.

7.1: IEFBR14 Utility

Example 1

- Commonly used to delete, allocate and to un-catalog dataset.
  - Example 1:

```
//DELETE        EXEC  PGM=IEFBR14
//* TO DELETE A FILE
//SYSPRINT      DD    SYSOUT=*
//DD1           DD      DSN=DA0001T.EMPLOYEE,
//                      DISP=(MOD,DELETE,DELETE),
//                      UNIT=SYSDA, SPACE=(TRK,0)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 4

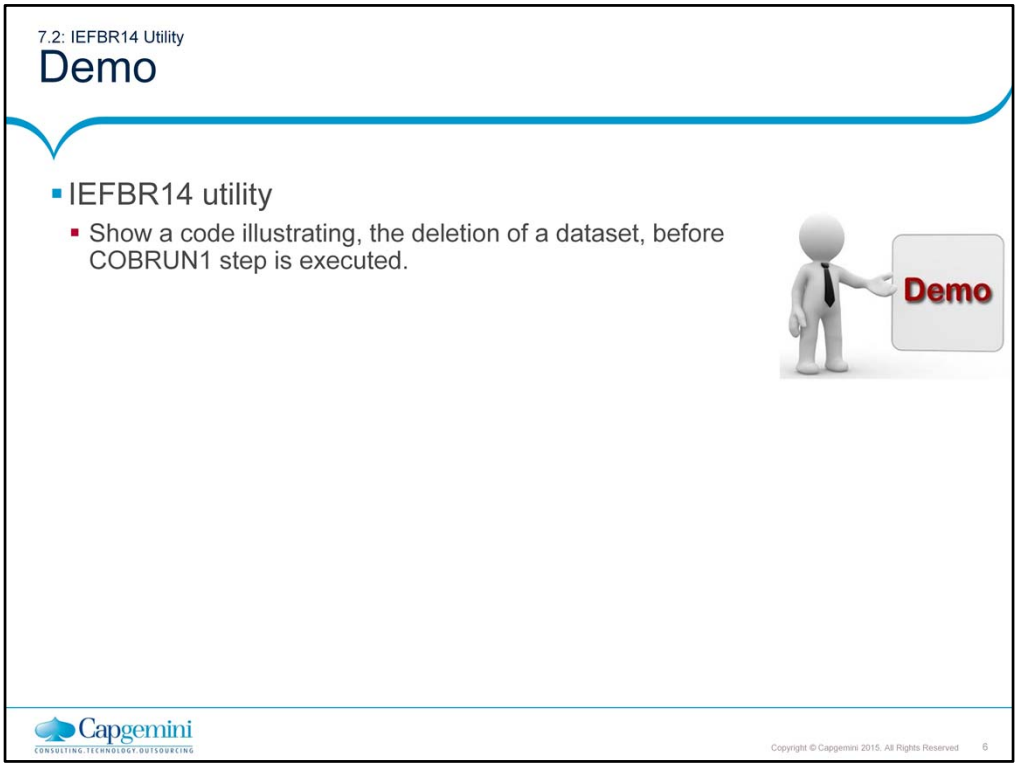
7.1: IEFBR14 Utility

Example 2

```
//CREATE        EXEC  PGM=IEFBR14
//* TO ALLOCATE A NEW FILE
//SYSPRINT      DD SYSOUT=*
//DD1           DD      DSN=DA0001T.EMPLOYEE,
//                      DISP=(NEW,CATLG,DELETE),
//                      UNIT=SYSDA, SPACE=(TRK,(2,1)),
//                      DCB=(BLKSIZE=800,LRECL=80,
//                      RECFM=FB,DSORG=PS)
```

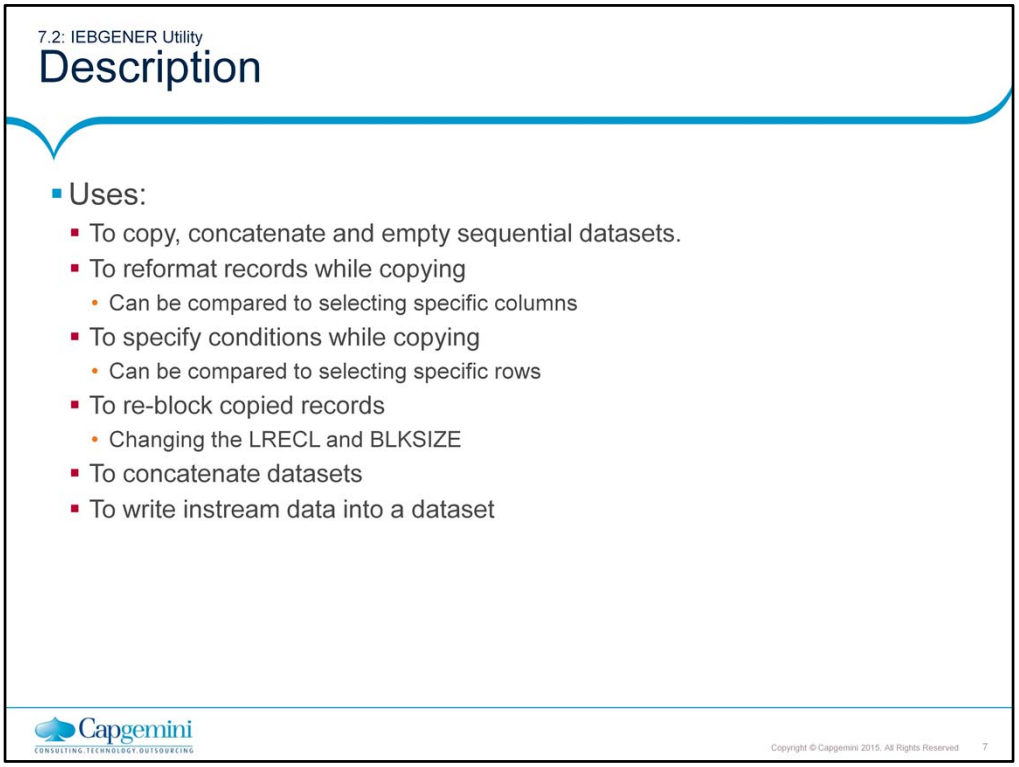
Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 5



An except of a code illustrating, the deletion of a dataset, before COBRUN1 step is executed.

```
//DA0001TA      JOB     LA2819,CG,NOTIFY=DA0001T,MSGCLASS=X,
MSGLEVEL=(1,0)
//DELETE        EXEC   PGM=IEFBR14
//SYSPRINT                      DD      SYSOUT=*
//LOGFILE       DD      DSN=DA0001T.MYFILE2,DISP=(MOD,DELETE,DELETE),
//    SPACE=(TRK,(0),),UNIT=SYSDA
//*
//COBRUN1       EXEC    PGM=ASS1, PARM='AAAA'
//STEPLIB       DD       DSN=DA00021T,PATNI.LOADLIB,DISP=SHR
//SYSPRINT      DD       SYSOUT=*
//INFILE        DD        DSN=DA00021T,EMPLOYEE,DISP=OLD
//OUTFILE       DD        DSN=DA00021T.MYFILE2,DISP=(NEW,CATLG,
DELETE),
//  DCB=(LRECL=80, DSORG=PS, BLKSIZE=80, RECFM=FB),
//   VOL=SER=BS3011, SPACE=(TRK, (45, 15))
//SYSOUT        DD      SYSOUT
//
```



To copy sequential datasets
Example

```
//STEP010     EXEC     PGM=IEBGENER
//SYSPRINT    DD       SYSOUT=*
//SYSUT1      DD       DSN=<Input dataset to be Read from>
//SYSUT2      DD       DSN=<Output dataset to be Written into>
//SYSIN       DD       DUMMY
//
```

The input dataset can be a PS or a member of PDS
The output dataset can also be a PS or a PDS member
Output datasets as PS apply to backing-up operations on tapes
The SYSPRINT DD statement defines the message dataset.
The SYSUT1 DD statement defines the input dataset.
The SYSUT2 DD statement defines the output dataset (can not have multiple SYSUT2).
The SYSIN DD statement defines the control dataset. This is where IEBGENER looks for any utility control statements. When DUMMY is specified, there are no control statements being used.
To copy sequential datasets

To copy an input sequential dataset to many members of a PDS Code Generate Maxname and Member name.

```
//DSRC012A    JOB      NOTIFY=DSRC012
//STEP010     EXEC     PGM=IEBGENER
//SYSPRINT    DD       SYSOUT=*
//SYSUT1      DD       DSN=<Input dataset>
//SYSUT2      DD       DSN=<Output PDS>
//SYSIN       DD       *
  Generate Maxname=3
  (To create 3 members in the output)
  Member name=(mem1,mem2,mem3)
  (mem1,mem2,mem3  will have the contents of sysut1)
/*
```

7.2: IEBGENER Utility

Sample

```
//DA0001TA  JOB          LA2719,CG,NOTIFY=DA0001T,
//                                  MSGCLASS=X
//*****************************************************************
//* USING THE IEBGENER Utility  TO EMPTY EXISTING  DATASET
//*****************************************************************
/CPYSTEP   EXEC PGM=IEBGENER
//SYSPRINT  DD          SYSOUT=*
//SYSUT1    DD          DUMMY, DCB=(BLKSIZE=800,
//                                  LRECL=80,RECFM=FB)
//SYSUT2    DD          DSN=DA0001T.MYOUT,DISP=OLD
//SYSIN     DD          DUMMY
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 8

7.2: IEBGENER Utility

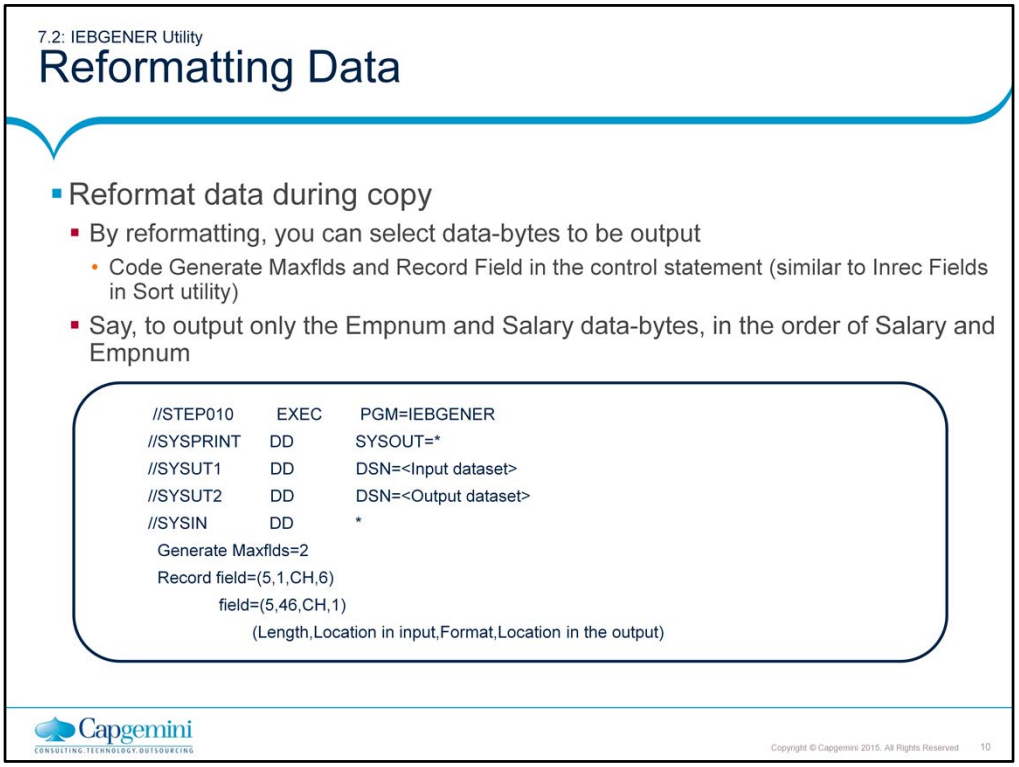
Demo

- IEBGENER Utility to merge the data from two sequential files
- To empty the existing dataset

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

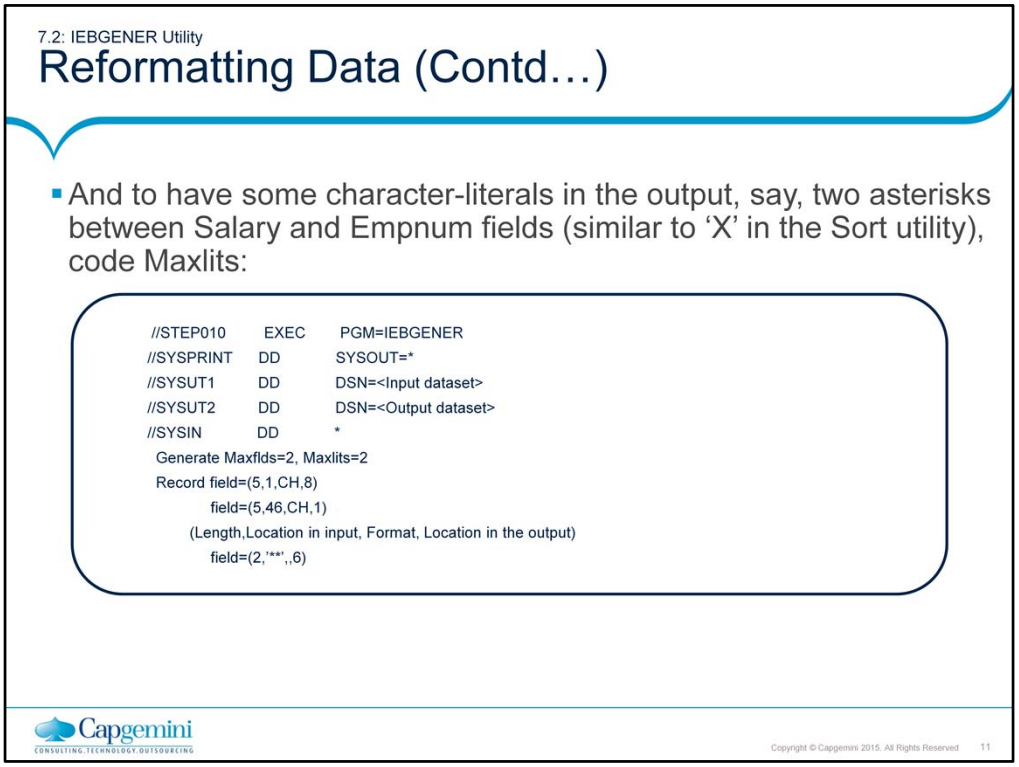
 9

```
//DA0001TA      JOB             LA2719, CG, NOTIFY=DA0001T,
//                                          MSGCLASS=X
//*************************************************************
//* USING THE IEBGENER Utility  TO MERGE DATASETS SYSUT1 PROVIDING
//* THE INPUT AND SYSUT2 BEING THE OUTPUT
//*************************************************************
//CPYSTEP       EXEC            PGM=IEBGENER
//SYSPRINT      DD              SYSOUT=*
//SYSUT1        DD              DSN=DA0001T.INDATA1, DISP=SHR
//SYSUT2        DD              DSN=DA0001T.NEW,DISP=MOD
//SYSIN         DD              DUMMY
//
```



Example:-

```
//JOBNAME  JOB NOTIFY=IGTRN30
//STEP01        EXEC PGM=IEBGENER
//SYSPRINT  DD SYSOUT=*
//SYSUT1     DD DSN=IGTRN30.CG.INFILE,DISP=SHR
//SYSUT2     DD DSN=IGTRN30.CG.OUTFILE,DISP=OLD
//*  LENGTH,START POSITION,,DESTINATION IN OUTFILE
//SYSIN       DD        *
GENERATE MAXFLDS=2
RECORD FIELD=(4,4,,10),FIELD=(10,4,,20)
/*
//
```



Example2:-

```
/IGTRN30A  JOB NOTIFY=IGTRN30
//STEP010   EXEC PGM=IEBGENER
//SYSPRINT  DD   SYSOUT=*
//SYSUT1    DD   DSN=IGTRN30.CG.INFILE,DISP=SHR
//SYSUT2    DD   DSN=IGTRN30.CG.PDS,DISP=OLD
//SYSIN     DD   *
  GENERATE MAXNAME=3
  MEMBER NAME=(MEM1,MEM2,MEM3)
/*
```

7.2: IEBGENER Utility

Concatenating Datasets

- To concatenate datasets

```
//DSRC012A   JOB   NOTIFY=DSRC012
//STEP010    EXEC  PGM=IEBGENER
//SYSPRINT   DD    SYSOUT=*
//SYSUT1     DD    DSN=<First unsorted dataset>
//           DD    DSN=<Second unsorted dataset>
//SYSUT2     DD    DSN=<The concatenated dataset>
//SYSIN      DD    DUMMY
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 12

7.2: IEBGENER Utility

Writing Instream Data

- To write instream data

```
 //DSRC012A    JOB        NOTIFY=DSRC012
//STEP010      EXEC       PGM=IEBGENER
//SYSPRINT     DD         SYSOUT=*
//SYSUT1       DD         *
  <Instream data>……
/*
//SYSUT2       DD         DSN=<Output dataset to be Written>
//SYSIN        DD         DUMMY
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 13

7.2: IEBGENER Utility

Demo

- IEBGENER Utility to copy the reformatted data
- IEBGENER Utility to concatenate the data
- IEBGENER Utility to write the instream data into a sequential file

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 14

7.3: IEBCOPY Utility

Description

- The IEBCOPY is used to copy members of partitioned datasets.
- The COPY statement identifies the input and output files by referring to their DDNAMEs in the JCL.
- The format is:
  - COPY OUTDD=output-DDname , INDD=input-Ddname
-

```
//MFCVT01A    JOB  NOTIFY=MFCVT01
//STEP1     EXEC PGM=IEBCOPY
//SYSPRINT  DD                SYSOUT=*
//IN        DD   DSN=MFCVT01.FILE1,DISP=SHR
//OUT       DD   DSN=MFCVT01.FILE2,DISP=SHR
//SYSIN     DD   *
COPY   OUTDD=OUT,INDD=IN
/*
```

- The above example copies all of the members from the PDS, 'MFCVT01.FILE1' to an existing PDS, 'MFCVT01.FILE2'.
- The IN and OUT DD statements define data sets to be used by IEBCOPY.
- The COPY control statement specifies the input and output DDNAMES.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 15

7.3: IEBCOPY Utility

Copying Members

- Copying Specific Members
  - The SELECT statement identifies the members of the PDS to be copied. The format is:
  - SELECT MEMBER=NAME (to specify a single member)

```
 //MFCVT01A    JOB  '0.2AMIP',.....
//STEP1      EXEC PGM=IEBCOPY
//SYSPRINT   DD              SYSOUT=*
//IN         DD   DSN= MFCVT01.FILE1,DISP=SHR
//OUT        DD   DSN= MFCVT01.FILE2,DISP=SHR
//SYSIN      DD   *
   COPY   OUTDD=OUT,INDD=IN
    SELECT MEMBER=ALLOCATE
/*
```

  - The above example copies the member called "ALLOCATE" from the PDS, 'MFCVT01.FILE1' to an existing PDS, 'MFCVT01.FILE2'.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 16

7.3: IEBCOPY Utility

Copying Members (Contd…)

- Copying Multiple Specific Members
  - SELECT MEMBER=(NAME,NAME,NAME) (to specify multiple members)

```
// MFCVT01A JOB  '0.2AMIP',.....
//STEP1     EXEC PGM=IEBCOPY
//SYSPRINT   DD  SYSOUT=*
//IN         DD   DSN= MFCVT01.FILE1,DISP=SHR
//OUT        DD   DSN= MFCVT01.FILE2,DISP=SHR
//SYSIN      DD   *
COPY   OUTDD=OUT,INDD=IN
SELECT MEMBER=(FILE1,FILE2,FILE3)
/*
```

  - The above example copies selected members FILE1, FILE2 and FILE3 from the PDS, 'MFCVT01.FILE1' to an existing PDS, 'MFCVT01.FILE2'.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 17

7.3: IEBCOPY Utility

Copying Members (Contd…)

- Copying and Renaming Specific Members

```
//MFCVT01A    JOB '0.2AMIP',.....
//STEP1       EXEC PGM=IEBCOPY
//SYSPRINT    DD                SYSOUT=*
//IN          DD  DSN=MFCVT01.FILE1,DISP=SHR
//OUT         DD  DSN=MFCVT01.FILE2,DISP=SHR
//SYSIN       DD  *
              COPY  OUTDD=OUT,INDD=IN
              SELECT MEMBER=(JOBA,(PROD,TEST,R))
/*
```

- The above example copies the member called “PROD” from the PDS, ‘MFCVT01.FILE1’ to an existing PDS, ‘MFCVT01.FILE2’.
- The SELECT control statement specifies:
- Copy the member JOBA
- the member PROD is to be copied in the following manner
  - rename PROD to TEST,
  - copy the renamed member TEST to the output dataset,
  - if a member by that name exists in the output dataset replace it.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 18

7.3: IEBCOPY Utility

Copying Members (Contd…)

- Copying Using EXCLUDE

```
//MFCVT01A    JOB  '0.2AMIP',.....
//STEP1      EXEC PGM=IEBCOPY
//SYSPRINT   DD         SYSOUT=*
//IN         DD   DSN=MFCVT01.FILE1,DISP=SHR
//OUT        DD   DSN=MFCVT01.FILE2,DISP=SHR
//SYSIN      DD   *
             COPY   OUTDD=OUT,INDD=IN
             EXCLUDE MEMBER=ALLOCATE
/*
```

- The above example copies all members 'MFCVT01.FILE1' except the member 'ALLOCATE'

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 19

7.3: IEBCOPY Utility

Copying Members (Contd…)

- Compressing Data Sets

```
//MFCVT01A    JOB  '0.2AMIP',.....
//STEP1      EXEC PGM=IEBCOPY
//SYSPRINT   DD              SYSOUT=*
//INPDS      DD   DSN=MFCVT01.FILE1,DISP=SHR
//SYSUT3     DD   UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSUT4     DD   UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSIN      DD   *
              COPY INDD=INPDS,OUTDD=INPDS
/*
```

- The above example compresses the library 'MTPL.FILE1'.
- Notice that the same DD name is specified in both the INDD and OUTDD parameters.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 20

7.2: IEBGENER Utility

Demo

- IEBGENER Utility to copy the partitioned dataset
- To copy the selected members
- Usage of exclude members
- Usage of compressing the dataset

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 21

7.4: IEHLIST Utility

Description

- The IEHLIST utility is used to
  - list entries in a DASD VTOC (Volume Table of Contents)
  - list entries in a PDS Directory.
  - list entries in a system catalog
- Example 1:

```
//STEP1    EXEC  PGM=IEHLIST
//SYSPRINT  DD     SYSOUT=*
//DD1       DD    DISP=OLD,UNIT=SYSDA,VOL=SER=ABC
//DD2       DD    DISP=OLD,UNIT=SYSDA,VOL=SER=DEF
//SYSIN     DD    *
     LISTVTOC  FORMAT,VOL=SYSDA=ABC
     LISTVTOC  FORMAT,VOL=SYSDA=DEF                    X
           DSNAME=(MTPL.FILE1,MTPL.FILE2)
/*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 22

7.4: IEHLIST Utility

Description (Contd…)

- The above example uses IEHLIST to print two VTOC listings:
- The IEHLIST looks for utility control statements coded below the SYSIN DD statements:
- The first LISTVTOC control statement requests an formatted (FORMAT) VTOC listing for pack ABC. This includes DSCB and space allocation information. If FORMAT is omitted, an abbreviated version is listed.
- The second LISTVTOC control statement requests a formatted VTOC listing for two datasets: MTPL.FILE1 and MTPL.FILE2.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 23

7.4: IEHLIST Utility

Description (Contd…)

- Example 2:

```
//STEP1    EXEC PGM=IEHLIST
//SYSPRINT  DD SYSOUT=*
//DD1      DD   DISP=OLD,UNIT=SYSDA,VOL=SER=ABC
//SYSIN    DD   *
      LISTPDS DSNAME=MTPL.FILE,VOL=SYSDA=ABC
/*
```

  - The above example uses IEHLIST to list entries in a PDS directory.
  - The LISTPDS control statement requests a listing of the directory for the PDS, MTPL.FILE.
- NOTE: DSNAME cannot be abbreviated as DSN on a control statement.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 24

7.5: SORT Utility

Description

- Provided by MVS
- Commonly used to:
  - sort data
  - copy selective data
  - remove duplicates
  - change data throughout the file
- Reorders Physical Sequential dataset as per requirement on given field(s).
  - These fields are called *control* or *key* fields.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 25

7.5: SORT Utility

Description (Contd…)

- Working
  - Assumes all input records to be out of sequence.
  - Puts them in a sequence you request.
    - Example: Employee data is sorted in the sequence of Emp. no., Emp.name or Salary etc.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 26

7.5: SORT Utility

The SORT Utility - DFSort

- To reformat data to be sorted
  - Can be compared to selecting specific columns
  - To specify conditions for selecting data
  - Can be compared to selecting specific rows
- To concatenate datasets and sort

```
//SYSIN    DD  *
SORT FIELDS=(1,5,CH,A)
/*
```

  - The control statement supplies a Sort-Key implying:
    - Start sorting the record at the absolute byte address (1),
    - Length, the number of bytes to be included in sorting (5) ,
    - Format of sorting (EBCDIC character),
    - Sequence of sorting (ascending / descending)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 27

7.5: SORT Utility

Format

```
Sort fields=(position, length, format, sequence)
```

- or

```
Sort fields=(position,length,sequence....),format=format
```

- Syntax is used if all fields on which the dataset to be sorted are of same type:
- Position: Location of input record's 1st byte of the key field
- Length: Length in bytes of the key field.
- Sum of all key fields (lengths) should not exceed 4092.
-

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 28

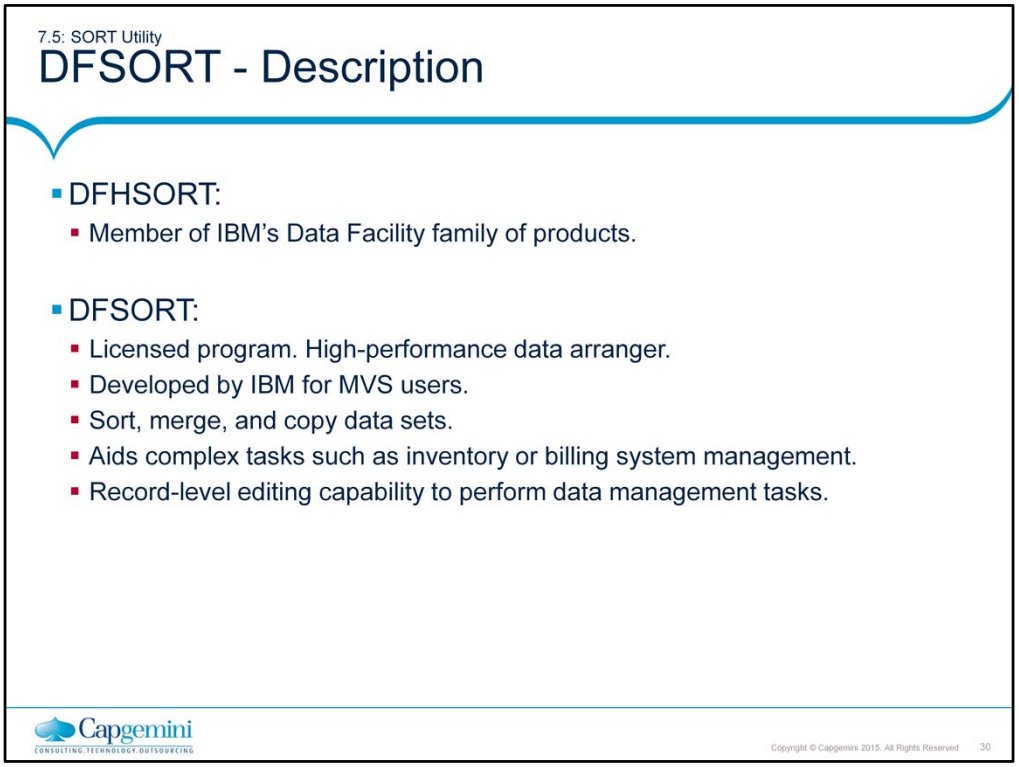
7.5: SORT Utility

Format (Contd…)

- Format: Two characters code identify the data format (type).
- Sequence:
  - A – Ascending
  - D - Descending

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 29



Sorting Data Sets
You can use DFSORT to rearrange the records in your datasets. Sorting is arranging records in either ascending or descending order within a file.

The fields in the records can be in any IBM System/370 format (for example EBCDIC character, decimal, and binary)

You can sort data in several different formats. Following table shows the most common data formats and the codes you use to specify them.

Data Format	**Code**
EBCIDIC (Character)	CH
Binary (Numeric)	BI
Zoned Decimal (Numeric)	ZD
Packed Decimal (Numeric)	PD

Merging Data Sets

You can also use DFSORT to merge data sets. DFSORT merges data sets by combining two or more files of sorted records to form a single data set of sorted records.

You can merge up to 16 data sets. The data sets you merge must be previously sorted into the same order (ascending or descending order).

The JCL needed for a merge is the same as that for a SORT, with the following exceptions:

- You do not use the SORTWKnn statement
- Instead of SORTIN DD statement, you use SORTINnn DD statements to define the input datasets. The SORTINnn DD statements name the input datasets to be merged and tell how many datasets are to be merged. The value nn in SORTINnn is a number from 0 to 16,indicating the number of datasets to be merged.

Copying Data Sets

DFSORT can also copy data sets without any sorting or merging taking place. You copy data sets in much the same way that you sort or merge them.

What else can you do with DFSORT?

While sorting, merging, or copying data sets, you can also:

- Select a subset of records from an input data set. You can include or omit records that meet specified criteria.
- Reformat records, add or delete fields, and insert blanks, constants, or binary zeros. For example, you can make a report more legible by inserting blank characters to separate fields.
- Sum the values in selected records while sorting or merging (but not while copying).
- Alter the collating sequence when sorting or merging records (but not while copying). For example, you can have the lowercase letters collate after the uppercase letters.

Creating and Running DFSORT Jobs

Processing data sets with DFSORT involves two steps:

1. Creating a DFSORT job
2. Running a DFSORT job.

You can run a DFSORT job by invoking processing in a number of ways stated as follows:

- With a JCL EXEC statement using the name of the program or the name of the cataloged procedure.
- With interactive panels supported under ISPF and ISMF.
- Within programs written in COBOL, PL1, or basic Assembler language.

Remarks: JCL-invoked means that the DFSORT program is initiated by JCL EXEC statement. The phrase dynamically invoked means that the DFSORT program is initiated from another program.

The JCL statements you need for most jobs are described as follows:

`//jobname JOB` Signals the beginning of a job.

`//stepname EXEC`

Signals the beginning of a job step and tells the operating system what program to run.

```
//stepname      EXEC PGM=SORT
```

`//STEPLIB DD`

defines the library containing DFSORT program. If your DFSORT program is in system library, you can omit the STEPLIB statement.

`//SYSOUT` DD defines the output data set for messages.

`//SORTIN` DD defines the input data set.

`//SORTWKnn` DD defines a work storage data set for a sort. For most applications, one work storage data set is sufficient. Increasing the number of work storage data sets does not improve performance.

`//SORTOUT` DD defines the output dataset.

`//SYSIN` DD Control statements.

All the control information within SYSIN DD can be coded freely between column 2 and column 71.

7.5: SORT Utility

SORT 1 JCL - Example

```
//DA0001T  JOB LA2719,CG, NOTIFY=DA0001T, MSGCLASS=X
//* SORT ON THE EMPLOYEE NAME IN ASCENDING //*ORDER
///SRTSTEP          EXEC     PGM=SORT
//SYSIN             DD   *
  SORT FIELDS=(1,5,CH,A)
/*
//SORTIN DD         DSN=DA0001T.EMPLOYEE,DISP=SHR
//SORTOUT           DD       DSN=DA0001T.OUTSORT,
//                           DISP=(NEW,CATLG, DELETE),
//                           SPACE=(TRK,(3,3)),UNIT=SYSDA
//SORTWK01    DD             SPACE=(TRK,(10,5)), UNIT=SYSALLDA
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 33

7.5: SORT Utility

SORT 1 JCL - Example (Contd…)

```
//SORTWK02    DD      SPACE=(TRK,(10,5)), UNIT=SYSALLDA
//SYSPRINT    DD      SYSOUT=*
//SYSOUT      DD      SYSOUT=*
//SORTMSG     DD      SYSOUT=*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 34

7.5: SORT Utility

SORT 1 JCL - Example (Contd…)

```
//STEP1    EXEC PGM=SORT
//SYSOUT   DD  SYSOUT=*
//SYSPRINT          DD          SYSOUT=A
//SORTIN   DD  DSN=MAINUSR.SEQ1.INPUT,DISP=OLD
//SORTOUT  DD  DSN=MAINUSR.SEQ2.OUTPUT,DISP=OLD
//SORTWK01  DD  UNIT=SYSDA,SPACE=(CYL,(20,10),RLSE)
//SYSIN    DD  *
            SORT FIELDS=(21,2,CH,A)
/*
```

- The above example will sort the records of the input dataset specified in the SORTIN DD statement based on the field specified in the control statement of the SYSIN DD. The sorted dataset is copied to the output dataset specified in the SORTOUT DD statement.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 35

7.5: SORT Utility

Demo

- SORT Utility (SORT JCL 1)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 36

7.5: SORT Utility

Sorting by Multiple fields - JCL 2

```
//DA0001T  JOB LA2719,CG,NOTIFY=DA0001T, MSGCLASS=X
//* SORTS  ON ASCENDING DEPTNO & DESCENDING ENAME
//SRTSTEP          EXEC     PGM=SORT
//SYSIN            DD   *
 SORT FIELDS=(17,2,PD,A,2,6,CH,D)
/*
//SORTIN DD        DSN=DA0001T.DEPT,DISP=SHR,
//                          SPACE=(TRK,(3,3)),UNIT=SYSDA
//SORTOUT     DD            DSN=DA0001T.SORTOUT2,
//          DISP=(NEW,CATLG, DELETE)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 37

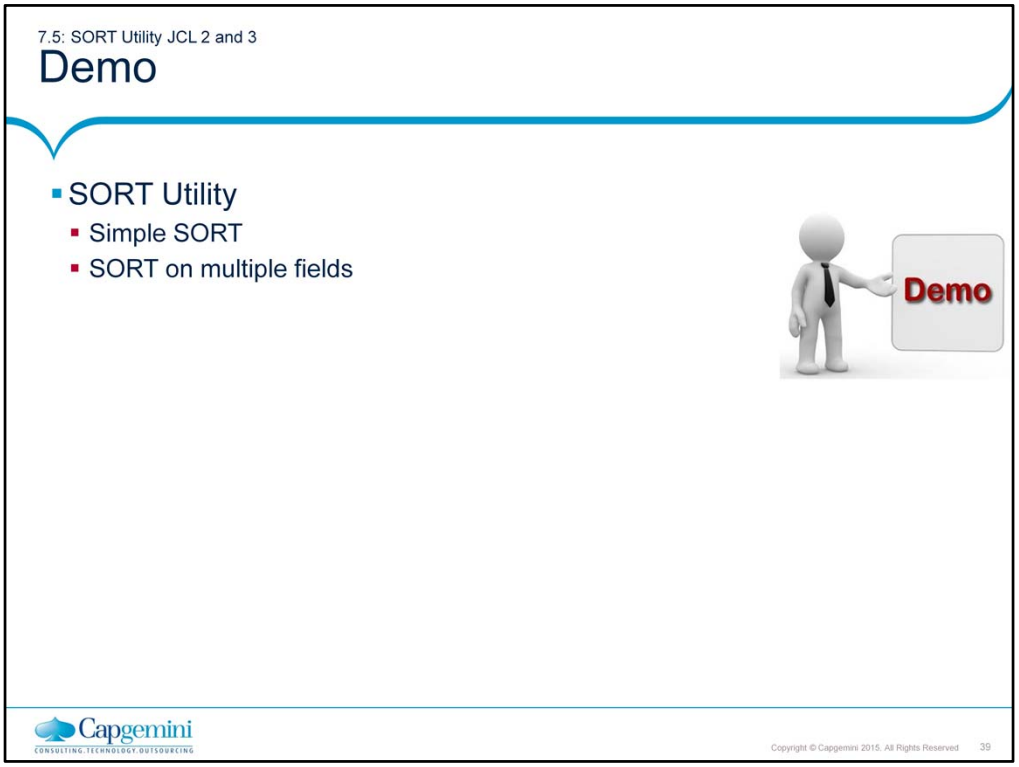
7.5: SORT Utility

Sorting by Multiple fields - JCL 2 (Contd…)

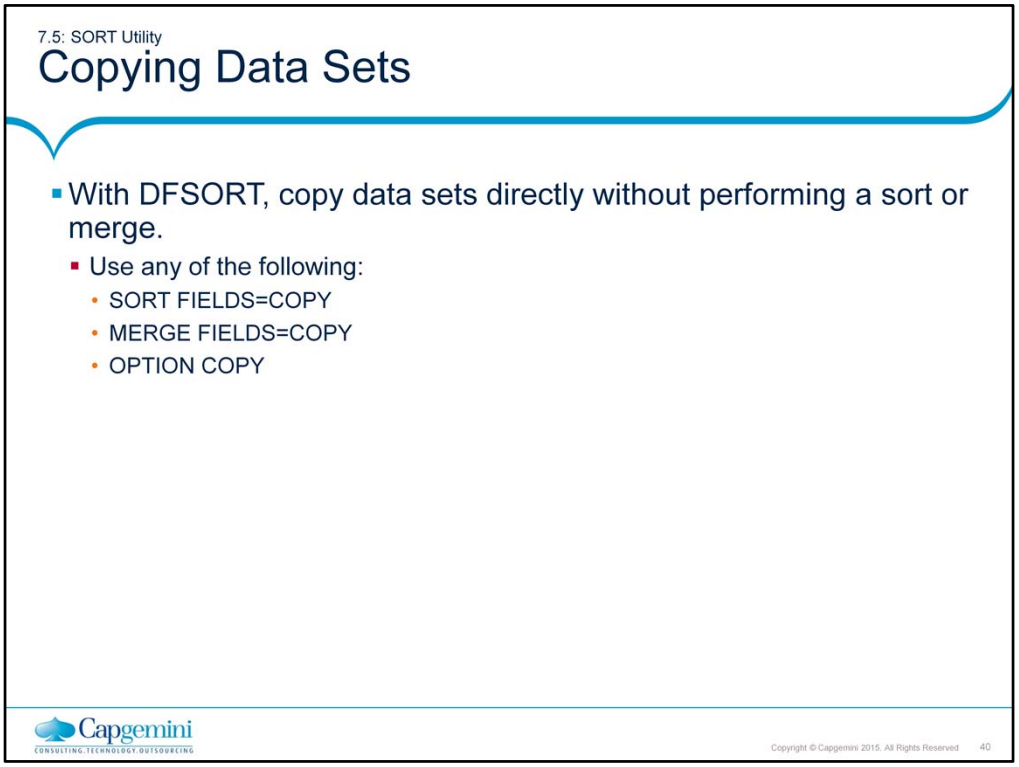
```
//SORTWK01    DD          SPACE=(TRK,(10,5)), UNIT=SYSALLDA
//SORTWK02    DD          SPACE=(TRK,(10,5)), UNIT=SYSALLDA
//SYSPRINT    DD          SYSOUT=*
//SYSOUT      DD          SYSOUT=*
//SORTMSG     DD    SYSOUT=*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

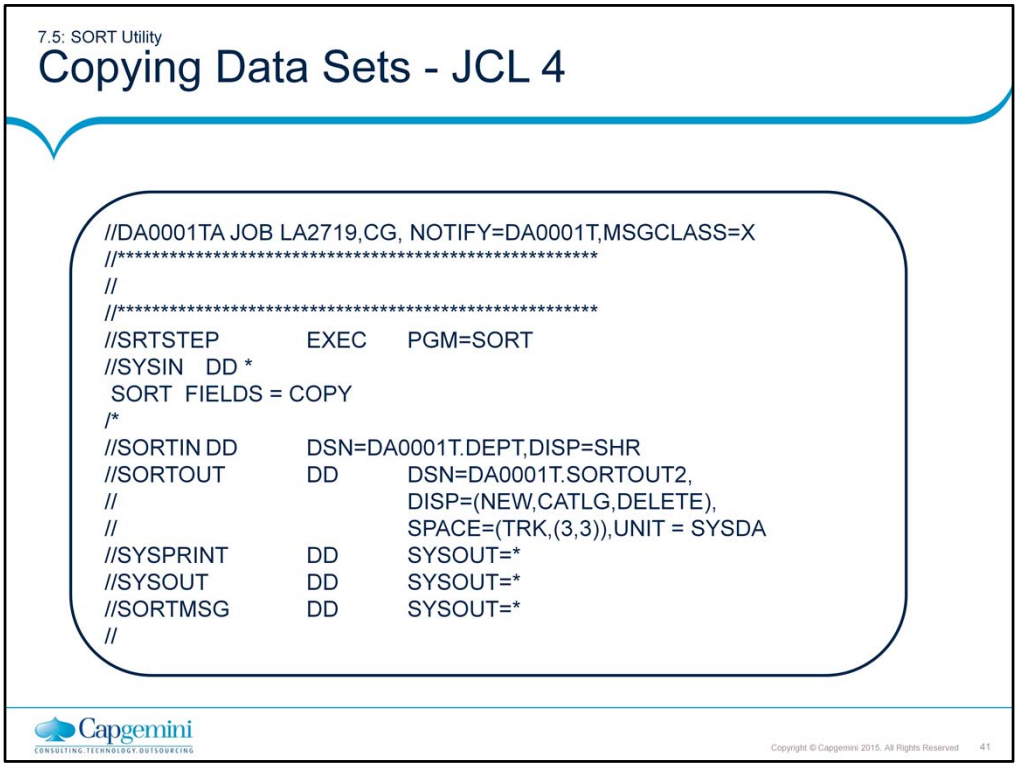
 38



You can further sort the records in the data set by specifying multiple control fields. When you specify two or more control fields, you specify them in the order of greater to lesser priority.



You can use COPY with all of the other DFSORT control statements except SUM. DFSORT can select and reformat the specific data sets you want to copy by using the control statements covered later.



The JCL for a copy application is the same as for a sort, except that you do not use the SORTWKnn DD statement.

7.5: SORT Utility JCL 4

Demo

- SORT Utility
  (SORT JCL 4)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 42

7.5: SORT Utility

Tailoring Input Data Set with INCLUDE and OMIT

- Tailor Data Sets:
  - You may need only a subset of the data set records for any application. Hence, you can tailor data sets.
  - Increase the speed of the sort, merge, or copy.
    - Fewer the records, lesser is the time taken to process them.
  - Steps to tailor an input data set:
    - Use an INCLUDE control statement to collect wanted records.
    - Use an OMIT control statement to exclude unwanted records.
    - Your choice of INCLUDE and OMIT depends on which is easier and more efficient to write for a given application.
      Note: INCLUDE and OMIT control statements are mutually exclusive.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 43

7.5: SORT Utility

Tailoring Input Data Set with INCLUDE and OMIT (Contd…)

- Select from the following comparison operators:

Comparison Operators	Meaning
EQ	Equal to
NE	Not Equal to
GT	Greater than
GE	Greater than or Equal to
LT	Less than
LE	Less than or equal to

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 44

7.5: SORT Utility

Rules to Tailor Input Data Set with INCLUDE and OMIT

- DFSORT uses following rules to pad and truncate strings.
  - Padding adds fillers in data, usually zeros or blanks.
  - Truncation deletes or omits leading or trailing portions.
  - Field-to-field Comparison: Shorter field is padded as appropriate (with blanks or zeros).
  - Field-to-Constant Comparison:
    - Constant is padded or truncated to the length of the field.
    - Decimal constants are padded or truncated on the left.
    - Character and hexadecimal constants are padded or truncated on the right.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 45

7.5: SORT Utility

Allowable Comparisons: INCLUDE and OMIT

- Following table shows field-to-field and field-to-constant comparisons:

Field Format	BI	CH	ZD	PD
BI	✔	✔		
CH	✔	✔		
ZD			✔	✔
PD			✔	✔

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 46

7.5: SORT Utility

SORT JCL 5 - Copy Selective Data

- INCLUDE COND copies data that matches a condition.
  - Example:
    - In this case it copies data with one character in the 19th position that equals 'M' or 'S'.

```
//DA0001TA JOB LA2719,CG,NOTIFY=DA0001T,MSGCLASS=X
//* SORTS  ON THE INPUT FILE ON JOB AND SELECTS JOB BEGINING
//*  WITH M OR S INTO A NEW DATASET
//SRTSTEP      EXEC  PGM=SORT
//SYSIN          DD   *
  OPTION  EQUALS
  SORT FIELDS=(19,6,A),FORMAT=CH
  INCLUDE COND=(19,1,CH,EQ,C'M',OR,19,1,CH,EQ,C'S')
/*
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 47

7.5: SORT Utility

SORT JCL 5 - Copy Selective Data (Contd…)

```
//SORTIN      DD  DSN=DA0001T.INDATA3,DISP=SHR
//SORTOUT     DD  DSN=DA0001T.SORTOUT3, DISP=(NEW,CATLG),
//               SPACE=(TRK,(3,3)), UNIT=SYSDA,
//               DCB=(BLKSIZE=800, LRECL=80,RECFM=FB,
//               DSORG=PS)
//SORTWK01    DD SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK02    DD  SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SYSOUT      DD  SYSOUT=*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 48

7.5: SORT Utility JCL 5

Demo

- SORT Utility (SORT JCL 5)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 49

7.5: SORT Utility

SORT 6 - Sort Selective Data

- Sort on Job.
  - Select jobs that begin with “M” and Deptno begins with **1**.

```
//DA0001TA  JOB LA2719,CG,NOTIFY=DA0001T,MSGCLASS=X
//* SORTS  ON JOB INCLUDES JOBS BEGINING WITH M AND  DEPTNO
//* BEGINNING WITH 1
//SRTSTEP      EXEC  PGM=SORT
//SYSIN          DD   *
  OPTION  EQUALS
  SORT FIELDS=(19,6,A),FORMAT=CH
  INCLUDE COND=(19,1,CH,EQ,C'M',AND,51,1,CSF,EQ,1)
/*
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 50

7.5: SORT Utility

SORT 6 - Sort Selective Data (Contd…)

```
//SORTIN DD   DSN=DA0001T.INDATA3,DISP=SHR
//SORTOUT        DD   DSN=DA0021T.SORTOUT4, DISP=(NEW,CATLG),
//                    SPACE=(TRK,(3,3)), UNIT=SYSDA,
//                    DCB=(BLKSIZE=800,LRECL=80,RECFM=FB,
//                    DSORG=PS)
//SORTWK01    DD  SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK02    DD  SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SYSOUT       DD  SYSOUT=*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 51

7.5: SORT Utility JCL 6

Demo

- SORT Utility (SORT JCL 6)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 52

7.5: SORT Utility

SORT JCL 7 - OMIT COND

- INCLUDE and OMIT are mutually exclusive.
  - Records that do not satisfy the condition are sorted and copied into the output dataset.
  - Example: Sorts on Job and omits Jobs that begin with M or S.

```
//DA0001TA  JOB LA2719,CG,NOTIFY=DA0001T,MSGCLASS=X
//* SORTS  ON JOB  OMITS JOBS BEGINNING WITH M OR S
//SRTSTEP           EXEC  PGM=SORT
//SYSIN             DD   *
  OPTION  EQUALS
  SORT FIELDS=(19,6,A),FORMAT=CH
  OMIT  COND=(19,1,CH,EQ,C'M',OR,19,1,CH,EQ,C'S')
/*
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 53

7.5: SORT Utility

SORT JCL 7 - OMIT COND (Contd…)

```
//SORTIN      DD  DSN=DA0001T.INDATA3,DISP=SHR
//SORTOUT     DD  DSN=DA0001T.SORTOUT5,DISP=(NEW,CATLG),
//                UNIT= SYSDA,SPACE=(TRK,(3,3)),
//                DCB=(BLKSIZE=800, LRECL=80,RECFM=FB,
//                DSORG=PS)
//SORTWK01    DD SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK02    DD SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK03    DD SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK04    DD SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SYSOUT      DD  SYSOUT=*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 54

7.5: SORT Utility JCL7

Demo

- SORT Utility (SORT JCL 7)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 55

7.5: SORT Utility

SORT JCL 8

```
//DA0001TA    JOB LA2719,CG, NOTIFY=DA0001T,MSGCLASS=X
//* Merges fields beginning with column 110 having length 5
//* INDATA1 and INDATA2 are sorted on the control field
//SRTSTEP          EXEC     PGM=SORT
//SORTIN01         DD  DSN=DA0001T.INDATA1,DISP=OLD
//SORTIN02         DD  DSN=DA0001T.INDATA2,DISP=OLD
//SORTOUT          DD  DSN=DA0001T.SORTOUT4,DISP=(NEW,CATLG),
//                     SPACE=(TRK,(3,3,)), UNIT = SYSDA,
//                     DCB=(BLKSIZE=800, LRECL=80,
//                     RECFM=FB,DSORG=PS)
//SYSOUT           DD  SYSOUT=*
//SYSIN   DD  *
 MERGE       FIELDS = (110,5,A),FORMAT=CH
/*
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 56

7.5: SORT Utility JCL 8

Demo

- SORT Utility (SORT JCL 8)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 57

7.5: SORT Utility

Writing Constants

- Formats to write character strings, hexadecimal strings and decimal numbers are as follows:
  - Character Strings
    - Format for writing a character string is: **C'x.....x'** where x is an EBCDIC character. For example, C'Sheela'.
    - If you wish to include a single apostrophe in the string, you must specify it as two single apostrophes. For example, O'NEILL must be specified as **C'O''NEILL'**.
  - Hexadecimal Strings
    - Format for writing a hexadecimal string is: **X'yy.....yy'** where yy is a pair of hexadecimal digits. For example X'C1C2' is equivalent to C'AB'.
  - Decimal Strings
    - Format for writing a decimal number is:
      n.....n or +n.....n or –n...n
      where n....n is a decimal digit. Examples are 24, +24, and –24.
    - Decimal number must not contain commas and decimal points.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 58

7.5: SORT Utility

Summing Records – SUM Statement

- Department TRG wishes to know the total salary of all trainers.
  - Use the INCLUDE statement to tailor the file to include only records for the TRG department.
  - Use SORT and SUM to get the sum of salaries.
- On the SUM control statement,
  - Specify one or more numeric fields to be summed whenever records have equal control fields.
    - Control fields are specified on the SORT statement.
  - Numeric fields: Binary, Packed Decimal, Zoned Decimal.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 59

7.5: SORT Utility

Summing Records – SUM Statement (Contd…)

- When you sum records, keep in mind that two types of fields are involved:
  - Control fields specified on the SORT statement.
  - Summary fields specified on the SUM statement.
- Writing the SUM Statement:

```
SUM FIELDS=(location, length, data-format,….)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 60

7.5: SORT Utility

Summing Records – SUM Statement (Contd…)

Example

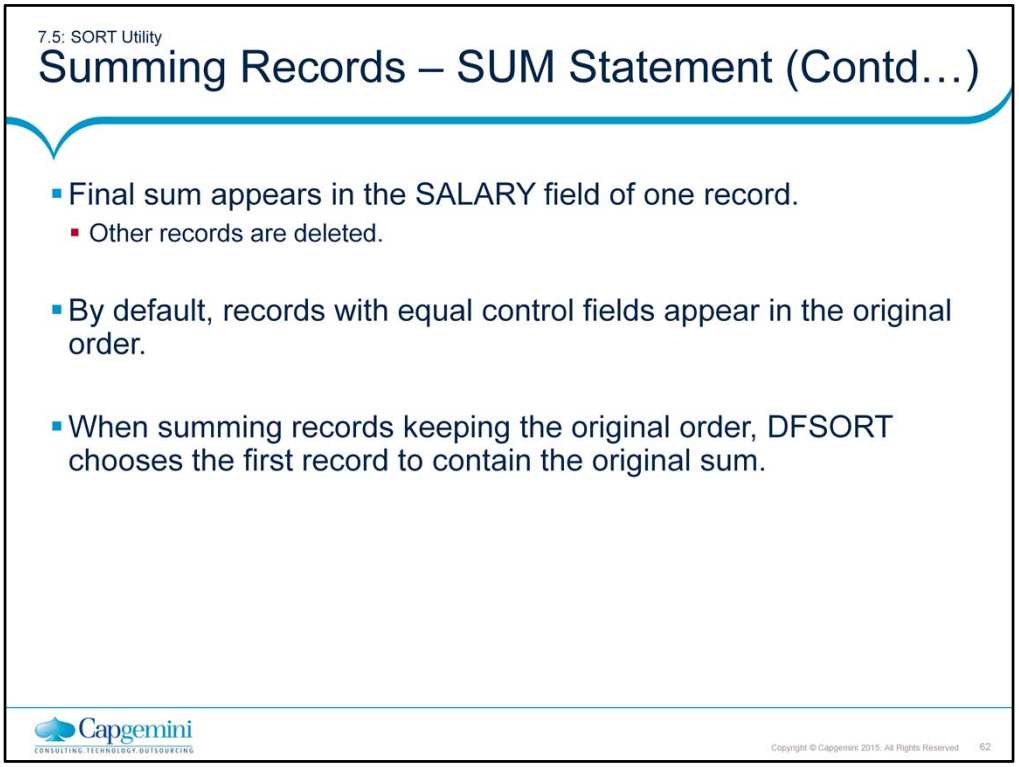
- INCLUDE, SORT, and SUM statements are shown below:

```
INCLUDE COND=(26,4,CH,EQ,C'TRG ')
SORT FIELDS=(26,4,CH,A)
SUM FIELDS=(35,5,BI)
```

- Returns the total salary of the TRG department.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 61



Remark: Some of the fields in your summation might not be meaningful, such as the employee number field. You could use the OMIT statement to omit this field. There are two other ways to leave out fields that are not meaningful.

7.5: SORT Utility

Suppress Records with Duplicate Control Fields

- Use SUM to delete records with duplicate control fields.
  - Specify FIELDS=NONE on the SUM statement.
  - Example: List all the distinct departments in ascending order.

```
SORT FIELDS=(25,4,CH,A)
SUM FIELDS=NONE
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 63

7.5: SORT Utility

Overflow

- Occurs when a sum becomes larger than the space available for it.
- If it occurs, the two records involved are left un summarized.
  - Contents of the records are left undisturbed
  - Neither record is deleted
  - Records are still available for summarization
- Does not prevent further summary
  - Correctable in some cases
  - Use INREC control statement to pad summary fields with zeros

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 64

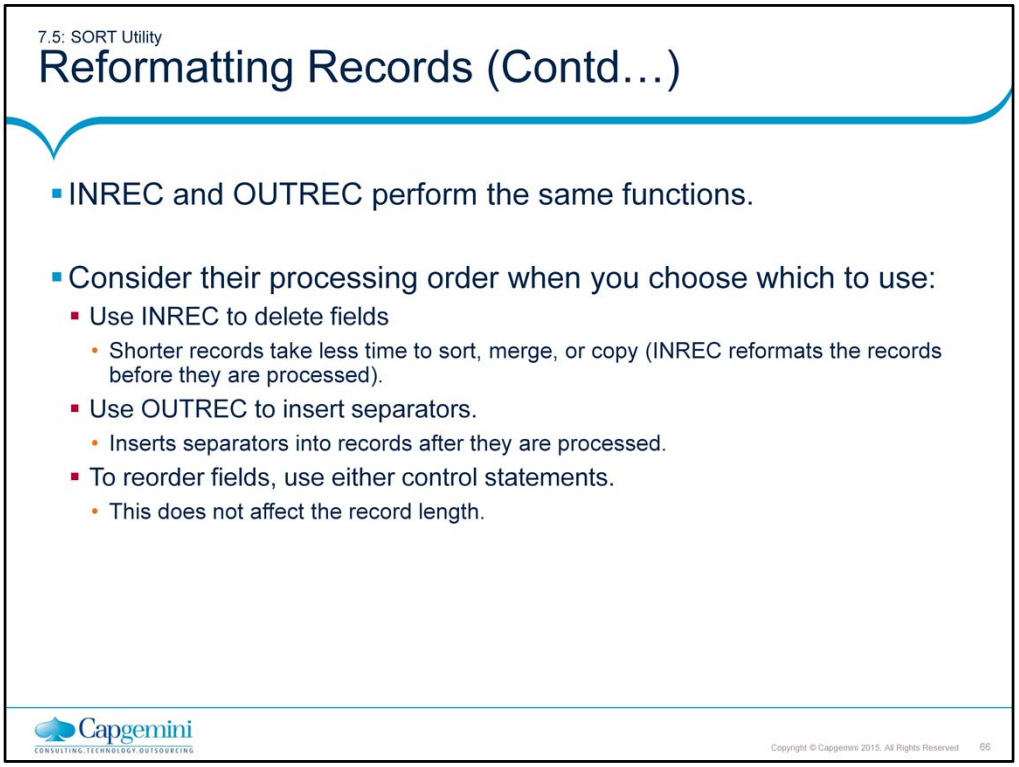
7.5: SORT Utility

Reformatting Records

- Reformat records in your data sets using OUTREC and INREC control statements
  - Delete fields
  - Reorder fields
  - Insert separators (blanks, zeros, or constants)
- Difference from DFSORT control statements:
  - OUTREC reformats records after they are sorted, copied, or merged
  - INREC reformats records before they are sorted, copied, or merged

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 65



Note: If you use INREC or OUTREC to change the record length, be sure to specify the final record length on the SORTOUT DD statement using the DCB parameter. The final length is either:

The INREC length if you are using just INREC.

The OUTREC length if you are using just OUTREC or both INREC and OUTREC.

7.5: SORT Utility

Reformatting Records - OUTREC

- Delete all unrequired fields for the application
  - Fields without meaningful contents in a summation record.
  - **Note:** In an OUTREC statement, you do not specify the data format.

```
SORT FIELDS=(26,4,CH,A)
SUM FIELDS=(35,5,BI)
OUTREC FIELDS=(26,4,35,5)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 67

7.5: SORT Utility

Reformatting Records - OUTREC (Contd…)

- As the record length changed, specify the new length on the SORTOUT DD statement.
  - For example:

```
//SORTOUT DD DSN=DA0001T.SORTOUT,
//           DISP=(NEW,CATLG,DELETE),
//           SPACE=(TRK,(1,1)),UNIT=SYSDA,
//            DCB=LRECL=9
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 68

7.5: SORT Utility

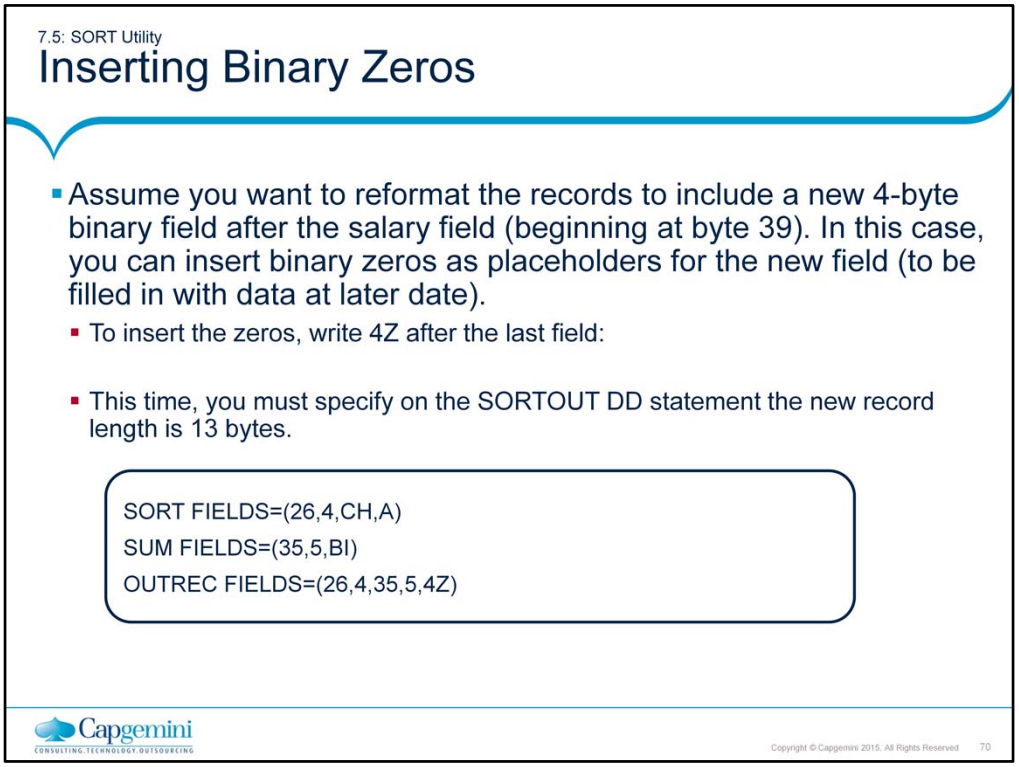
Reorder Fields to Reserve Space

- Fields always appear in the order in which you specify them.
- Therefore, if you wish for salary to appear before department, simply reverse the order in the OUTREC statement.

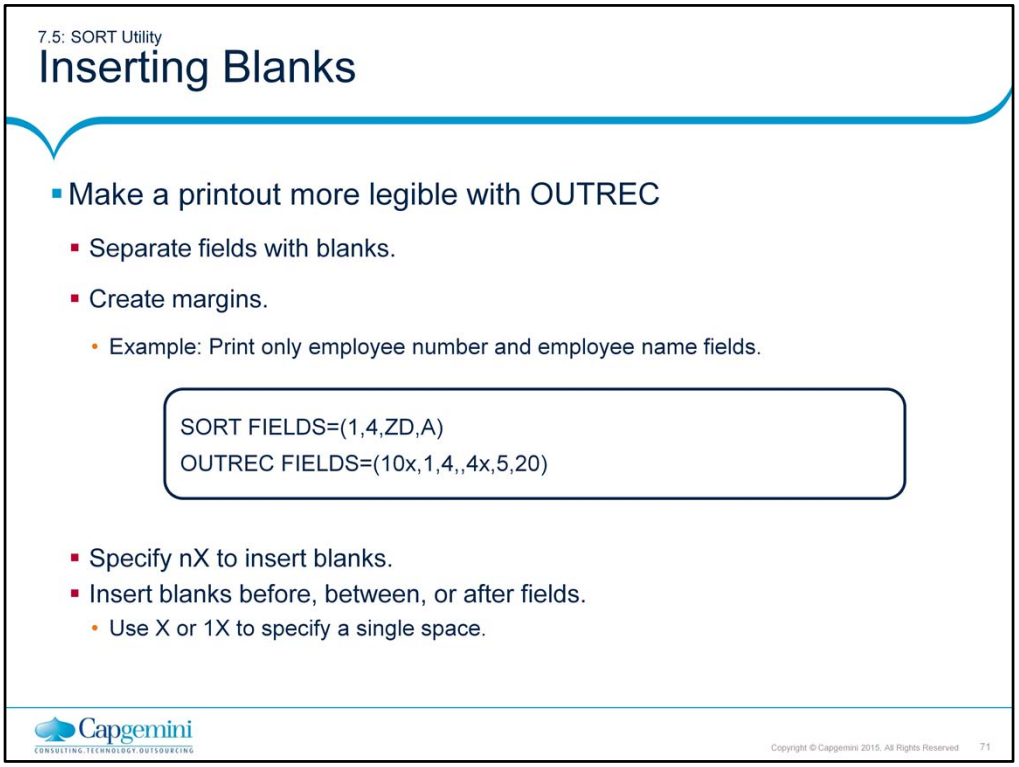
```
SORT FIELDS=(26,4,CH,A)
SUM FIELDS=(35,5,BI)
OUTREC FIELDS=(35,5,26,4)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 69



You can insert binary zeros before, between, or after fields. You can use Z or 1Z to specify a single binary zero.



If an output data set contains only character data, you can print it by writing the SORTOUT DD statement as follows:

```
//SORTOUT DD SYSOUT=*
```

7.5: SORT Utility

Inserting Constants

- Using OUTREC, insert constants to set up a report format.
- Formats to write constants are shown below:
  - Character Strings
    - Format to write a character string is: C'x…..x' where x is an EBCDIC character. For example, C'Sheela'.
    - Format to write a character string repetition is: nC'x…..x'

Where n can be from 1 to 4095; n repetitions of the character string constant (C'x…x') are inserted into the reformatted input records. If n is omitted, 1 is used instead.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 72

7.5: SORT Utility

Inserting Constants (Contd…)

- If you want to include a single apostrophe in the string, you must specify it as two single apostrophes. For example, O'NEILL must be specified as C'O''NEILL'.
  - Hexadecimal Strings
    - Format for writing a hexadecimal string is: X'yy……yy' where yy is a pair of hexadecimal digits. For example X'C1C2 is equivalent to C'AB'.
    - Format to a hexadecimal string repetition is: nC'yy…..yy'

Where n can be from 1 to 4095; n repetitions of the hexadecimal string constant X'yy…yy') are inserted into the reformatted input records. If n is omitted, 1 is used instead.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 73

7.5: SORT Utility

Setting up the Report Format - Example

- Following statement sets up the report as shown below:

```
OPTION COPY
 OUTREC FIELDS=(11:C'THE EMPLOYEE NUMBER IS ',1,4,
     30:C'THE EMPLOYEE NAME IS ',5,20,4X,25,4)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 74

7.5: SORT Utility

Reformat Records Using INREC Statement

- INREC statement has the same format as OUTREC

```
INREC FIELDS=(26,4,35,5)
➢ SORT FIELDS=(1,4,4,CH,A)
SUM FIELDS=(5,5,BI)
```

- As INREC reformats records before they are sorted.
  - SORT and SUM statements must refer to the reformatted records as they appear in the output data set.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 75

7.5: SORT Utility

Preventing Overflow Summing Values

- Prevent overflow in some cases using INREC to pad summary fields with zeros.
- However, you cannot use this method for negative fixed-point binary data.
  - Padding with zeros rather than ones would change the sign.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 76

7.5: SORT Utility

Padding Summary fields

- If summary fields overflow, you can pad each of them on the left with 4 bytes (binary fields must be 2, 4, or 8 bytes long).

```
INREC FIELDS=(26,4,4Z,35,5)
SORT  FIELDS=(1,4,CH,A)
SUM   FIELDS=(5,10,BI)
```

- You cannot use the OUTREC statement to prevent overflow, as it is processed after summarization.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 77

7.5: SORT Utility

Processing Order Of Control Statements

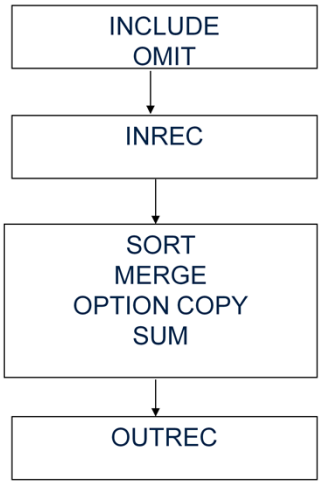
- Subsequent flowchart shows the order in which control statements are processed.
  - SUM is processed at the same time as SORT or MERGE.
  - It is not used with COPY.
- You can write statements in any order.
- However, DFSORT always processes them in the order shown as follows.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 78

7.5: SORT Utility

Processing Order Of Control Statements - Flow Chart



Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 79

7.5: SORT Utility

SORT JCL 9 - Example

- Sorts on salary removing duplicates, includes only salaries > 2000.
  - To sort and pick up selective data, remove duplicates.
  - SUM FIELDS=NONE is used to remove duplicates.
  - It compares data in columns mentioned in SORT FIELDS= and removes second occurrence of matching data.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 80

7.5: SORT Utility

SORT JCL 9 - Example (Contd…)

```
//DA0001TA        JOB LA2819,CG,NOTIFY=DA0001T,MSGCLASS=X
//SRTSTEP         EXEC  PGM=SORT
//SYSIN           DD   *
  OPTION  EQUALS
  SORT FIELDS=(41,4,A),FORMAT=CSF
  INCLUDE COND=(41,4,CSF,GT,2000)
  SUM FIELDS=NONE
/*
//SORTIN DD       DSN=DA0001T.INDATA3,DISP=SHR
//SORTOUT         DD      DSN=DA0001T.SORTOUT6,DISP=(NEW,CATLG),
//                        SPACE=(TRK,(3,3)), UNIT=SYSDA,
//                        DCB=(BLKSIZE=800,LRECL=80,
//                        RECFM=FB,DSORG=PS)
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 81

7.5: SORT Utility

SORT JCL 9 - Example (Contd…)

```
//SORTWK01    DD        SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK02    DD        SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK03    DD        SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SORTWK04    DD        SPACE=(TRK,(10,5)),UNIT=SYSALLDA
//SYSOUT          DD    SYSOUT=*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 82

7.5: SORT Utility JCL 9

Demo

- SORT Utility (SORT JCL 9)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 83

7.5: SORT Utility

SORT JCL 10 - Example

```
//DA0001TA  JOB LA2719,CG,NOTIFY=DA0001T, MSGCLASS=X
//*Copies those records where either employee jobs begin with 'M' or 'S'
//SRTSTEP          EXEC     PGM=SORT
//SYSOUT           DD       SYSOUT=*
//SORTIN           DD       DSN=DA0021T.INDATA3,DISP=SHR
//SORTOUT          DD       DSN=DA0021T.SORTOUT7,
//                          DISP=(,CATLG,DELETE), UNIT=SYSDA,
//                          SPACE=(TRK,(5, 2))
//
SYSIN              DD   *
  OPTION  COPY
  INCLUDE COND=(19,1,CH,EQ,C'M',OR,19,1,CH,EQ,C'S')
/*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 84

7.5: SORT Utility JCL 10

Demo

- SORT Utility (SORT JCL 10)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 85

7.5: SORT Utility

SORT JCL 11 - Example

- Change data throughout file.
  - In this example, the 163rd character in the file is changed to C.
  - Useful to change data in a file which is more than 255 characters in length as TSO edit option cannot be used for it.

7.5: SORT Utility

SORT JCL 11 - Example (Contd…)

```
//DA0001TA     JOB LA2719,CG,NOTIFY=DA0001T, MSGCLASS=X
//SRTSTEP      EXEC  PGM=SORT
///SYSOUT            DD   SYSOUT=*
//SORTIN             DD   DSN=DA0001T.EMPLOYEE,DISP=SHR
//SORTOUT            DD   DSN=DA0001T.EMPLOYEE,DISP=SHR
//SYSIN              DD   *
OPTION  COPY
OUTREC FIELDS =(1,162,C'C',164,137)
/*
//
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 87

7.5: SORT Utility JCL 11

Demo

- SORT Utility (SORT JCL 11)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 88

7.5: SORT Utility

Reformatting Code

- To reformat, code:
  - Before sort fields, INREC FIELDS to reformat the record before sorting
  - After sort fields, OUTREC FIELDS to reformat the record after sorting
- Consider the following layout of a PS with fixed length records:
  - Empno :starting at absolute byte 1, for 5 bytes long
  - Empname :at absolute byte 6, for 25 bytes long
  - Department :at absolute byte 31, for 15 bytes long
  - Salary :at byte 46, for 5 bytes long
  - The PS being populated with the following records:
    - 11111sujit admin 10000
    - 55555danny marketing 15000
    - 22222ajay admin 07000
    - 33333mala projects 20000^

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

7.5: SORT Utility

Reformatting Code (Contd…)

- Before sorting, to include only employee number and salary field and to sort in descending order of salary field:

```
//SYSIN    DD    *
INREC FIELDS=(1,5,46,5)----------1111110000
SORT FIELDS=(6,5,CH,D)              5555515000
/*                                  2222207000
                                    3333320000
```

- The sorted dataset will contain:

3333320000
5555515000
1111110000
2222207000

7.5: SORT Utility

Reformatting Data-Bytes

- To reformat data-bytes after sorting, OUTREC FIELDS is used.
  - Also used to space out fields in the output.
  - To obtain only the Empno and Salary fields in the output, but to sort on Empname field in ascending order:

```
//SYSIN        DD        *

  SORT FIELDS=(6,25,CH,A) ---------- 22222ajay.....
OUTREC FIELDS=(1,5,46,5)             55555danny.....
                                     33333mala.....
                                     11111sujith.....
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 91

7.5: SORT Utility

Reformatting Data-Bytes (Contd…)

- To include character-literals, say, spaces between the Empno and Salary fields in the output dataset:
  - Code OUTREC FIELDS=(1,5,2X,46,5)

Two spaces

- The output dataset will contain:

```
22222  07000
55555  15000
33333  20000
11111  10000
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

7.5: SORT Utility

SORT JCL 12 - Example

```
DA0001TA    JOB LA2719,CG, NOTIFY=DA0001T,MSGCLASS=X
//* Merges fields beginning with column 110 having length 5
//* INDATA1 and INDATA2 are sorted on the control field
//SRTSTEP          EXEC     PGM=SORT
//SORTIN01         DD  DSN=DA0001T.INDATA1,DISP=OLD
//SORTIN02         DD  DSN=DA0001T.INDATA2,DISP=OLD
//SORTOUT          DD  DSN=DA0001T.SORTOUT4,DISP=(NEW,CATLG),
//                     SPACE=(TRK,(3,3,)), UNIT = SYSDA,
//                     DCB=(BLKSIZE=800, LRECL=80,
//                     RECFM=FB,DSORG=PS)
//SYSOUT           DD  SYSOUT=*
//SYSIN   DD  *
 MERGE       FIELDS = (110,5,A),FORMAT=CH
/*
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 93

7.5: SORT Utility JCL 12

Demo

- SORT Utility (SORT JCL 12)

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 94

7.5: SORT Utility

Example

- To select employees getting a salary of 1000 and sort in ascending order of employee number

```
//SYSIN        DD        *
INCLUDE COND=(46,5,CH,EQ,C'1000)
SORT FIELDS=(1,3,CH,A)
/*
```

- Conversely, Omit Cond can be used to exclude employees with salary of 1000.
- INCLUDE and OMIT are mutually exclusive
- Can use connect operators AND and OR to form several logical conditions

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 95

7.5: Lab

Lab

- Day 3 and Day 4 Labs

Summary

- IEFBR14 is commonly used to delete, allocate and un-catalog a dataset.
- IEBGENER Utility is commonly used to copy, concatenate and to empty sequential datasets.
- SORT Utility is used to sort data, copy selective data, remove duplicates, and change data throughout the file.

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 97

Review Question

- Question 1: Which of the following utility is used to uncatalog a dataset?
  - IEBGENER
  - IEFBR14
  - SORT
- Question 2: Which of the following utility can be used to copy contents of a sequential dataset?
  - IEFBR14
  - IEBGENER
  - SORT

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

 98