



Bansilal Ramnath Agarwal Charitable
Trust's Vishwakarma Institute of
Information Technology

Department of

Name: Yuvraj Prabhakar Sankilwar

Class: TY

Division: A

Roll No: 391052

Semester: VI

Academic Year: 2024-2025

Subject Name & Code: ECI3120G - Mainframe Technologies

Title of Assignment: 1. Write a JCL SORT

Date of Performance: 23/03/2025

Date of Submission: 25/03/2025

Aim:

Manually create a PS (Physical Sequential) file using ISPF 3.2 option containing 100 records, each 80 bytes in length. The records are structured such that the primary key is located between columns 13 and 20. Out of the 100 records, 10 are duplicates, and the remaining 90 are unique. The records are stored in random order.

Problem Statement:

Task: Write a JCL SORT to:

- Eliminate the duplicate records.
- Sort the remaining records based on the primary key (columns 13-20).

Background Information:

In enterprise computing environments, IBM mainframes have long served as the backbone of critical business applications and large-scale data processing. One of the fundamental ways data is stored and manipulated in mainframe systems is through datasets, particularly sequential datasets.

1. Introduction to Sequential Datasets

A **Physical Sequential (PS)** dataset is a dataset where data is stored in a sequential manner, meaning records are stored one after another on disk, and must be accessed in the same sequence. These datasets are commonly used in batch processing jobs where data needs to be read and processed from start to end without random access.

In this assignment, a PS dataset is used to store **100 records**, each exactly **80 bytes** in length. This uniform length simplifies the processing logic, especially when using utilities like **DFSORT** for sorting and deduplication. Each record contains a **primary key** located between **columns 13 and 20**, which will serve as the basis for sorting and identifying duplicates.

2. Role of JCL in Mainframe Job Execution

JCL (Job Control Language) is a scripting language used on IBM mainframes to instruct the system on how to run batch jobs. It defines:

- The programs or utilities to be executed (like SORT),
- The datasets to be used (input, output),
- Resource requirements such as memory or disk space,
- Disposition of datasets after job completion (whether to delete, catalog, or keep them).

A JCL job consists of multiple statements, typically including:

- A **JOB statement**, which defines the job's name and accounting info,
 - An **EXEC statement**, which identifies the program or utility to execute,
 - One or more **DD (Data Definition) statements**, which define the input/output datasets.
-

3. Introduction to DFSORT Utility

DFSORT is one of the most powerful and commonly used utilities in z/OS environments. It provides robust support for:

- **Sorting** data in ascending or descending order,
- **Eliminating duplicate records** using `SUM FIELDS=NONE`,
- **Filtering** data based on conditions (`INCLUDE/OMIT`),
- **Merging** multiple datasets,
- **Summarizing** data.

In the context of this assignment, DFSORT is being used for two main purposes:

a. Sorting the Dataset by Primary Key

The primary key lies between columns **13 and 20**, which means the sorting needs to be based on **8 characters** starting from position 13. The DFSORT statement used is:

```
SORT FIELDS=(13,8,CH,A)
```

This sorts the records in **ascending alphanumeric order** (CH means character format, A means ascending).

b. Removing Duplicate Records

To remove duplicates based on the **entire record**, the following DFSORT control statement is

used:

SUM FIELDS=NONE

This tells DFSORT to suppress duplicate records that are identical in content, keeping only the first occurrence.

4. Dataset Design and Key Characteristics

For this assignment:

- The input dataset is named Z66845.INPUT.PS, a PS file containing 100 fixed-length records.
- Each record is 80 bytes long and contains structured data, with the **primary key in columns 13–20**.
- The records are inserted in **random order**, and **10 out of 100** records are duplicates.
- The output dataset is Z66845.OUTPUT.PS, which will contain only the **unique records**, sorted by the primary key.

The process simulates a **real-world data cleansing task**, where raw data often contains redundancy and needs to be normalized and ordered before loading into databases or further analysis.

5. Practical Importance of Sorting and Deduplication

In enterprise scenarios, sorting and removing duplicates is an essential step in preparing datasets for:

- **Data warehousing** or **ETL (Extract, Transform, Load)** operations,
- **Master data management**, where data from multiple sources must be consolidated,
- **Report generation**, where only distinct records are needed,
- Ensuring data **integrity and consistency** before processing transactions.

By leveraging DFSORT in JCL, mainframe systems efficiently handle large volumes of data without requiring custom code. This significantly reduces processing time and resource usage.

GitHub Repo Link:

<https://github.com/yuvrajofficials/mainframe-assignments.git>

Conclusion:

This assignment not only provides hands-on experience with JCL and DFSORT, but also illustrates foundational concepts in mainframe data processing. Understanding how to define, sort, and clean datasets using batch jobs is critical for any aspiring mainframe developer or system programmer. Mastery of these tools ensures efficient data handling in mission-critical applications that run on z/OS systems.