Lab 8. Working with COMP-3

Note: All action items followed by a * sign, include exercise hints, which are located at the end of this exercise.

Overview

Many of the previous COBOL lab programs you have worked with thus far are reading records containing two packed decimal fields, the client account limit and the client account balance. In this lab, the total of all client account limits and balances used a COMPUTE statement, where the COMP-3 fields contained the packed decimal internal data.

Objectives

- Practice the use of COMPUTE
- Understand the need for COMP-3 fields
- Identify the compile error and understand the underlying cause
- Use that information to correct and verify the program runs successfully

Preface

- What happens when an internal packed decimal field is not described using COMP-3?

  Without using COMP-3 to describe the field, the COBOL program treats the data as DISPLAY data (EBCDIC format). This lab demonstrates what happens during program execution without using COMP-3.

Using VSCode and Zowe Explorer:

1. Submit the job, **id.JCL(CBL0010J)**

2. Observe that the compile of the COBOL source was successful, however, also observe that the execution of the job failed. How can you tell?

   There's no CC code next to CBL0010J(JOB#), instead there is an ABENDU4038 message. U4038 is a common user code error typically involving a mismatch between the external data and the COBOL representation of the data.

3. Read the execution SYSOUT message carefully. The SYSOUT message mistakenly believes the records are 174 characters in length while the program believes the records are 170 characters in length.

   Packed decimal (COMP-3) expands into two numbers where only one number would typically exist. If the program reads a packed decimal field without describing the field as COMP-3, then program execution becomes confused about the size of the record because the PIC clause, S9(7)V99, is expecting to store seven numbers plus a sign digit when only three word positions are read. Therefore, execution reports a four-record length position discrepancy.

4. Edit id.CBL(CBL0010) to identify and correct the source code problem.*

5. Submit id.JCL(CBL0010J) and verify correction is successful with a CC 0000 code.

Lab Hints

4. The ACCT-LIMIT PIC clause in the ACCT-FIELDS paragraph should be the same as the PIC clause for ACCT-BALANCE.