INFO3105 Week 9 Part 1

PCS2PRG1 Cont'd… - Remember you need to submit a CLEAN compile 1st on your PCS2PRG1 program, then submit JCS2LDRN .

Important NOTE:

➔ STARTING with Lab12 onwards until the end of this course, I will allow/encourage you to work with a partner on this development work (max 2 people) and I will give both partners the same mark. Note however that we will be learning a number of concepts during these labs which will show up on the Final exam (20%), so careful to plan for sharing the learning/understanding parts of these labs…

Processing the Remaining Transaction Types

In the last class we looked at the ADD and DELETE transactions, now we'll look at the remaining transaction types. Remember to determine the transaction types you should be using the EVALUATE statement something like this:

```
EVALUATE TRUE
    WHEN TRANS-ADD
        MOVE TRANS-DATA TO WS-TRANS-MAINTENANCE
        PERFORM C100-PROCESS-ADD
    WHEN TRANS-DEL
        MOVE TRANS-DATA TO WS-TRANS-MAINTENANCE
        PERFORM C200-PROCESS-DEL
    WHEN TRANS-CHG
        MOVE TRANS-DATA TO WS-TRANS-MAINTENANCE
        PERFORM C300-PROCESS-CHG
    WHEN TRANS-SALE
        MOVE TRANS-DATA TO WS-TRANS-SALE
        PERFORM C400-PROCESS-SALE
    WHEN TRANS-RET
        MOVE TRANS-DATA TO WS-TRANS-RETURN
        PERFORM C500-PROCESS-RETURN
 END-EVALUATE
```

Processing UPDATE/Change Transactions

Similar to the Add, the **Update** process moves the data from the transaction file to the master file. An update would be used to change things like department, branch or name information. However, we can't do a WRITE here or else a new record will be attempted to be added and since the key already exists an error will occur. So what we need to issue in this case is a **READ, followed by some updates (ie. moves to update the appropriate fields/variables), then a REWRITE** statement as follows (in depth look at the REWRITE can be found on pages **408-409** of the text):

```
READ SALESMAST
  INVALID KEY MOVE 'PROBLEM GETTING MASTER RECORD'
    TO  WS-ER-PROBLEM
END-READ.
```

... some moves to update the appropriate fields/variables ... then:

```
REWRITE SALESPERSON-MASTER
  INVALID KEY MOVE 'PROBLEM REWRITING MASTER STATUS IS:'
                                  TO  WS-ER-PROBLEM
END-REWRITE.
```

To help you with Salesperson-Master processing, just like you did for the previous lab, you will also need to include the status checking Cobol code after the READ, REWRITE, WRITE, DELETE ... so **for each** of these updates (or reads) you will need to include code like (or similar to) this:

```
IF  WS-IN-STATUS = '00'  OR  WS-IN-STATUS = '02'

    *   02 STATUS MEANS DUPLICATE WRITTEN TO AIX WHICH IS OK

        PERFORM  W110-PRINT-TRANSACTION-LINE

ELSE

        MOVE  TRANS-NO  TO  WS-ER-TRN-NO

        MOVE  'PROBLEM DOING xxxxx  STATUS IS:'  TO  WS-ER-PROBLEM

        MOVE  WS-IN-STATUS  TO  WS-ER-IDX-STATUS

        WRITE  PRNT-REC  FROM  WS-ERROR-LINE

END-IF.
```

Processing SALES Transactions

Processing Sales transactions involve both a **READ** and a **REWRITE**, and it differs from an UPDATE transaction because we need to update the financial data not the demographic fields/data. We need to **READ** the existing Year-to-Date value - salesperson YTD gross sales before we can update it with the new sales information by updating the YTD gross sales amount. Before doing so, you need to calculate what the sale actually is. The sale can be calculated by taking the transaction sales amount and **applying the discount** % to it (remember to **use ROUNDED** here). So the transaction gross sales amount calculation would be Salesamount - Salesamount*discount%. Once the transaction sale amount is calculated, add it to the current YTD Gross Sales Amount and then rewrite the record to update the salesperson YTD gross sales.

An example from our data is often most enlightening:

SALES TRANSACTION LOG

TRANS NO.	SALESPERSON	TRANS DATE	TYPE	AMOUNT
1	19304	13/11/01	SALE	$4,974.96

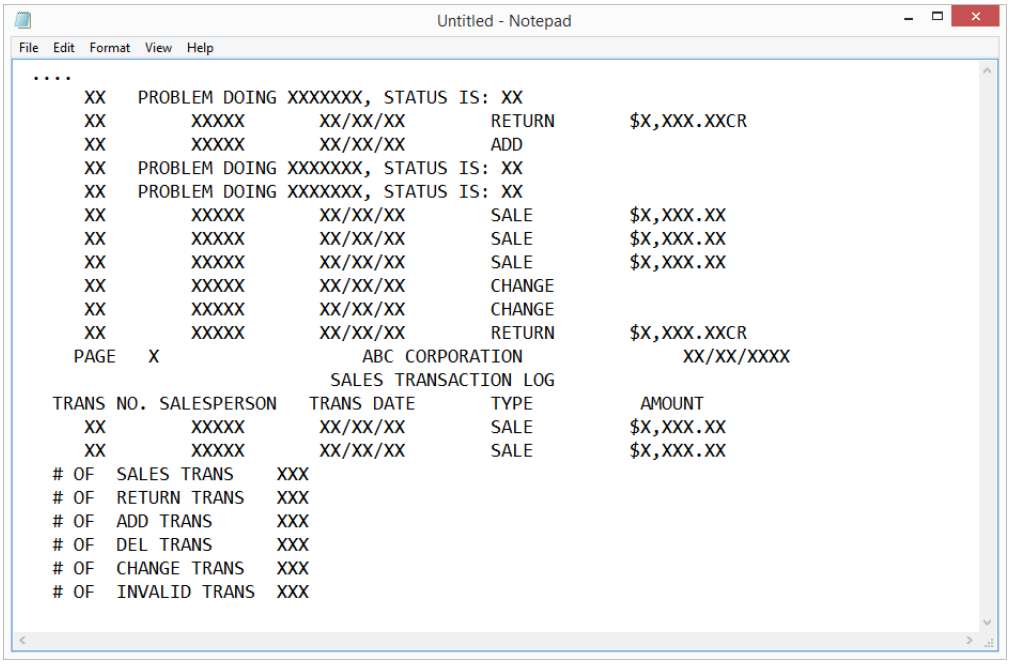
So for Salesperson No 19304 (Cavaretta, Katherine), she started with $ 63,009.61 in Gross sales before these transactions were processed, plus this $4,974.96 sale ($6,218.70 from the SALESTRANS record, minus 20% discount) ... and so after this update should have total Gross sales of $67,984.57

Also, notice that there are a **few transactions that are in error** you will have to accommodate for that by checking the status of the Reads, Writes, and Re-writes and record the number of invalid transactions.

Processing RETURN Transactions

Similar to the Sales transaction, except here you are updating the Return amount not the Gross sales amount and you only have to add the return amount to the YTD total.

Completing the Transaction Log



Here is some sample output of a transaction log. The actual numbers have been blanked out on purpose as you typically won't have that luxury when programming in a real scenario. (A Hint though ... if we had a large group of friends that were thirsty on a hot summer day, we might ask for the last two # OF CHANGE TRANS XXX – and # OF INVALID TRANS ... ie. I'll have a ## of Labatts Blue at the beer store ...).

The only "new" concept here is the edited field for returns, eg:

```
05  WS-TL-AMT                    PIC $$$,$$$.99CR.
```

The **CR** is typically used to designate money removed from an account and is to be used on the return lines (remember from Lab 4 how we got the CR to show...) .

Also, some of the transactions have invalid data in them (hey it happens in real life) so we need to flag which transactions are invalid as well, see above "**PROBLEM DOING**…." You'll need to count those as well see the last line of the report.

Lab 12 - 2%

This is the final part of this program. To summarize, create the report and the amount column again pay particular attention to get numbers aligned and formatted properly.

Submit both as .txt files, non-zipped please :

- Complete compile and link jcl results for PCS2PRG1
- Complete JCS2LDRN results