Y4T –YYYYMMDD

Y4W-MMDDYYYY

F1,DATADIFF,F2

F1 ,ADDDAYS,+12,TOGREG=Y4W(/)

ADDMONS

ADDYEARS

SUBDAYS

SUBMONS

SUBYEARS

Arrays

01 ws-table

02 ws-marks pic 9(5) occurs 5 times

Subscript or index variable or hard core

02 ws-marks occurs t times indexed by ws- i(indexed)

01 ws- sub pic 9(03)

Move 75 to ws-marks(1).

Move 67 to ws –marks(2).

Subscript

01 ws-sub pic ((notesubscpit range 1 to 5) no 0, -1 )

01 ws-item pic 9(03)

Move 1 to ws-sub

Move 75 to ws-sub(ws-item)

Add 1 to ws-item

Move 80 to ws-sub(ws-item)

Indexed.

01 ws-mark

05 ws-sub 9(03) occurs 5 times indexed by sub-index

Set sub-index to 10

Move 75 to ws-sub(sub-index)

Set sub-index up by 1

Accept ws-mark

Move ws-mark to ws-sub(sub-index)

Perform 5 items

Accept ws-mark

Add 1 to ws –item.

Move ws-mark to ws-sub(ws-items)

End-perfrom.

01 ws-dep.mgr occrs 5 times.

05 mgr pic x(05)

05 dep pic x(05).

If dep(ws-item) =ws-acpt dept.

Display mgr(ws-item)

Search table

At end display “not found’

When dep(mgr) =ws-acpt-mgr

Display mgr(index)

End –search.

Pefrom para1 until eof

Pefrom para12 until end-ofarray.

end-para.

Read file .

Validation

String handling:

String

Unstring

Inspect.

Name=’raja’ str2=raja .str1=mr.

Mr.raja

‘mr:’name.

String (identifier or iteral) delimited by size or space into string

String str1 delimited by size str2 delimited by space into str3.

String str1 delimted by’.’

Str2 delimted by’j’

Into str3

Mrra

Note :move option remaining with blanks in string option direcly copy.

On overflow.

Overflow greater than destination length.

On size error

Display ‘error size’

End-string.

Unstring.

Unstring identifier 1 delimited by into two varibles

Inspect

1. Inspect variable tallying counting variable for all ‘x’. Or (ws-str2)

2. Inspect variable replacing all ‘x’ by ‘a’.

3. Inspect ws-str3 tallying w-cout for leading w-star after w-str2.

4. Converting ‘abc’ to ‘xyz’.

**Usage class (comp variables)**

Move length of emp1 to len.

IN COBOL usage class represents the memory allocation of data

In usage we have the following types

1.comp(computational) comp is binary format which contain sum defines ranges

18byptes in 8 bytes

9(4) 2 bytes

9(8) 4 bytes

2.comp1 is single precision floating point varible

Comp1 doesnot contain any picture clause because length of comp1 fixed for 4 bytes

Comp2 double precision floating point numer no picture clause fixed size 8 bytes and store 18 charcters of data

Comp3

Comp3 a packed decimal will store 2 characters of data into 1 bytes memory

The length of any data item can be caluculated below formula

1 .picture clause even (n/2)+1

1. odd (n+1/2

In general usage class we have default value “DISPLAY’

USAGE DISPLAY ACTS AS TO print the data of data item declare under data division.

1.Add pgm program

IDENTIFICATION DIVISION.

PROGRAM-ID. ADDPGM1.

ENVIRONMENT DIVISION.

DATA DIVISION.

WORKING-STORAGE SECTION.

01 WS-NO1 PIC 9(2).

01 WS-NO2 PIC 9(2).

01 WS-RES PIC 9(2).

PROCEDURE DIVISION.

0000-MAIN-PARA.

ACCEPT WS-NO1.

ACCEPT WS-NO2.

COMPUTE WS-RES = WS-NO1 + WS-NO2.

DISPLAY 'THE NUMBER 1:' WS-NO1.

DISPLAY 'THE NUMBER 2:' WS-NO2.

DISPLAY 'THE RESULT IS:' WS-RES.

STOP RUN.