BCI433 - IBM i Business Computing

Week 4.5: CL Programming with Display Files

Agenda

- CL Programming
- Lab 4-2 (Part E, G) is not required for Summer 2020

Lesson Objectives

Upon completion of this lecture and lab 4-2 you'll be able to:

- Create interactive CL program with display file

CL Programming Restrictions

- Only five *FILE per program
  - Display file or Database File
- Can’t update Database Files
- Can’t create reports

- Note: CL programs are used to run and manage business applications (RPGLE) as batch and interactive jobs that utilize resources efficiently

CL Variables

- Data types
  - *CHAR , *DEC , *LGL, *INT, *UINT
- Variable names start with an '&'
  - e.g. &IN03, &CTR, &USER, &DATE, &MARK1, etc.
- Declare CL Variable (DCL)
  - e.g. DCL VAR(&varname) TYPE(*CHAR) LEN(8)
  - variables must be declared before you can use them
  - variables from a display file will automatically be available to program.
- Change Variable value
  - e.g. CHGVAR VAR(&varname) VALUE(value)

Examples

```
DCL  VAR(&TOTAL) TYPE(*DEC) LEN(7  2)
DCL VAR(&GRADE) TYPE(*CHAR) LEN(1)
DCL  &ABLE  *DEC LEN(5 2)
DCL &CHAR *CHAR LEN(10)

CHGVAR  VAR(&GRADE)  VALUE (‘A’)
CHGVAR  VAR(&TOTAL)   VALUE(&TOTAL + 1)
```

Operators

- Arithmetic (+, -, *, /)
- Character (*CAT, ||, *BCAT, |>, *TCAT, |<)
- Relational (*EQ, =, *GT, >, *LT, <, *GE, >=, *LE, <=, *NE, *NG, *NL)
- logical (*AND, *OR, and *NOT)

Concatenating Strings

- *CAT joins together two strings.
- *BCAT strips out the trailing blanks of the first string and then inserts one blank space between the first and second strings.
- *TCAT first strips out the trailing blanks of the first string, then joins that with the second string. The second string is not touched.
- It helps to remember:
  - CAT is which by keeping in mind **B for blanks** and **T for truncate.**

Concatenating Strings *CAT

- Example code:

```
DCL &F1 *CHAR 10 ‘IBC’
DCL VAR(&F2) TYPE(*CHAR) LEN(10) VALUE(‘233’)
DCL &F3 *CHAR 20

CHGVAR &F3 (&F1 *CAT  &F2)
```

- What will &F3 have in it?

Concatenating Strings *BCAT

- Example code:

```
DCL &F1 *CHAR 10 ‘Hello’
DCL VAR(&F2) TYPE(*CHAR) LEN(10)
   VALUE(‘World!’)
DCL &F3 *CHAR 20

CHGVAR &F3 (&F1 *BCAT  &F2)
```

- What will &F3 have in it?

Concatenating Strings *TCAT

- Example code:

```
DCL &F1 *CHAR 10 ‘IBC  ’
DCL VAR(&F2) TYPE(*CHAR) LEN(10) VALUE(‘233’)
DCL &F3 *CHAR 20

CHGVAR &F3 (&F1 *TCAT  &F2)
```

- What will &F3 have in it?

Some File Commands

- **DCLF** - Declares a File
  - (files must be declared before you can use them)
  - e.g. DCLF FILE(MARKSDF)

    or DCLF MARKSDF

- **SNDRCVF** - Sends a record to a screen and waits for the user to enter input, then reads it
  - (only for Display Files) –
  - used with input/output screens
  - e.g. SNDRCVF(MARKSDF)

  - How to make a field to be read only?
    - In display file, set the field's display attribute to protected – DSPATR(PR)

Condition and Iteration

- IF (condition) THEN(command) ELSE
  - for conditions, *AND, *OR, *NOT, *LT, *GT, *EQ, *NL, *NG, etc
- DO and ENDDO
  - used when you need to execute several commands in an IF statement
  - = { … } in C, Java, …
- GOTO **and** labels

IF, THEN, ELSE examples

- e.g. 1

```
IF   COND(&TIME   *LE   120000)  +
     THEN( SNDMSG   MSG('Good Morning')  TOUSR(DC233A40)
)
ELSE                                          +
     CMD( SNDMSG MSG('Good Afternoon')  TOUSR(DC233A40 ) )
```

- e.g. 2

```
IF   (&TIME   *LE   120000)  +
     THEN( SNDMSG   'Good Morning'  DS233A36)
ELSE                                          +
      (SNDMSG    'Good Afternoon'   DS233A36)
```

IF, THEN, ELSE examples

- e.g. 3

```
IF   (&A = &B)  THEN(DO)
          CHGVAR   VAR(&IN32) VALUE('1')
          CHGVAR   VAR(&IN33) VALUE(‘0')
     ENDDO
ELSE DO
          CHGVAR   VAR(&IN32) VALUE('0')
          CHGVAR   VAR(&IN33) VALUE(‘1')
    ENDDO
```

- e.g. 4

```
IF   (&A = &B)  DO          /* Without THEN() */
          CHGVAR   VAR(&IN32) VALUE('1')
          CHGVAR   VAR(&IN33) VALUE(‘0')
      ENDDO
```

DO, ENDDO example

```
IF   (&CHOICE = 'O'    *OR    &CHOICE = 'o')    +
     DO
        CHGCURLIB  IBC233LIB
        WRKOBJPDM  IBC233LIB
     ENDDO
ELSE (GOTO END)

END:  ENDPGM
```

DOWHILE Loop

```
SNDRCVF

DOWHILE(&IN03 *NE ‘1’)
   IF (&IN05    *EQ   ‘1’)  +
     DO
        CHGVAR   VAR(&MARK1)   VALUE(0)
        CHGVAR  VAR(&MARK2)    VALUE(0)
     ENDDO
  SNDRCVF
ENDDO

WRKOBJPDM
```

This bit of code sends the display file to the screen and reads it back. If F3 is pressed, the loop exits and WRKOBJPDM is done. If not, It then checks to see if the user has pressed F5. If so, it initializes the 2 fields MARK1 and MARK2 and redisplays the screen.

DOUNTIL Loop

```
SNDRCVF

DOUNTIL(&IN03)       /* the contents of loop is */
  IF (&IN05) +           /* always processed once */
   DO
     CHGVAR   VAR(&MARK1)   VALUE(0)
     CHGVAR  VAR(&MARK2)    VALUE(0)
   ENDDO
  SNDRCVF
ENDDO

WRKOBJPDM
```

This bit of code sends the display file to the screen and reads it back. If F3 is pressed, the loop exits and WRKOBJPDM is done. If not, It then checks to see if the user has pressed F5. If so, it initializes the 2 fields MARK1 and MARK2 and redisplays the screen.

Select Group

```
SELECT
   WHEN COND(&TYPE *EQ *CMD) THEN(DO)
        (group of CL commands)
   ENDDO
   WHEN COND(&TYPE = *PGM) THEN(DO)
        (group of CL commands)
   ENDDO
   OTHERWISE CMD(CHGVAR &BADTYPE ‘1’)
ENDSELECT
```

CL Subroutines

- Execute a subroutine
  - e.g. CALLSUBR INIT
- Define a subroutine

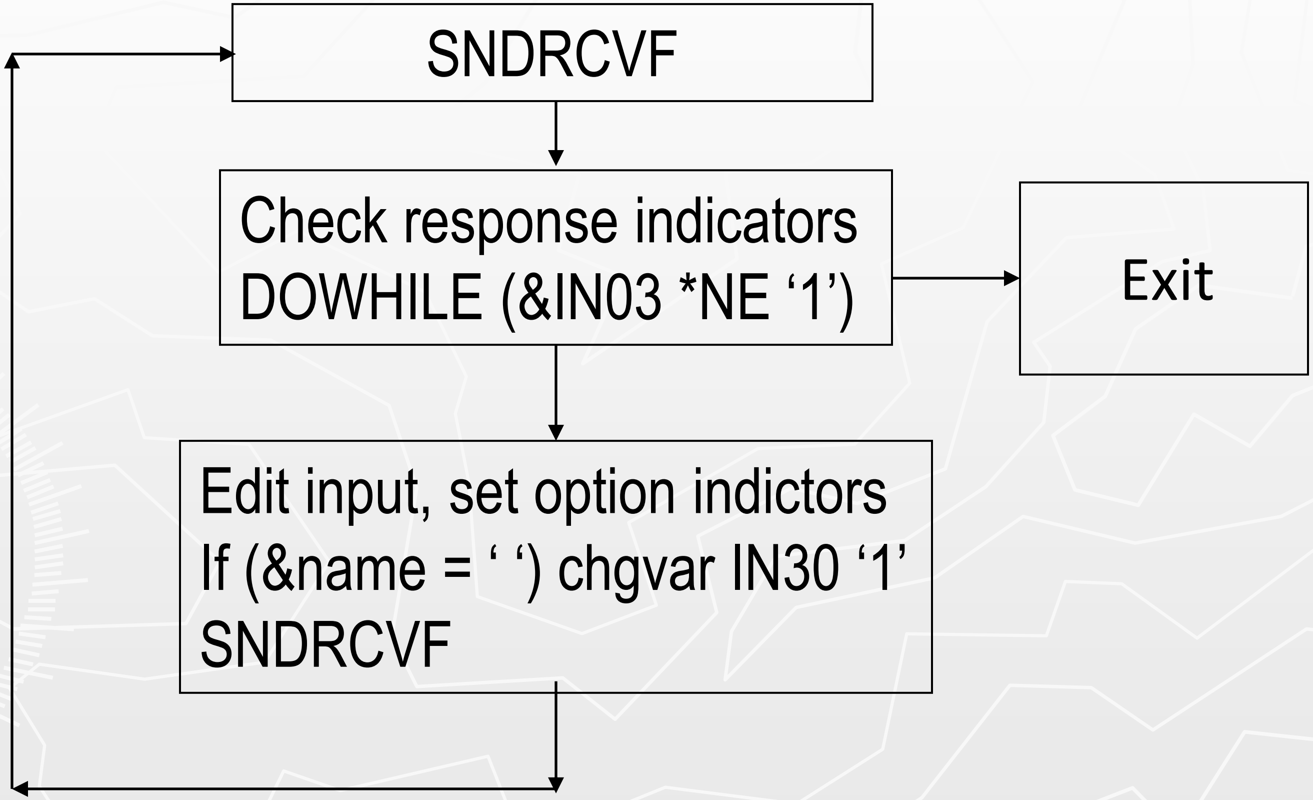
e.g.

```
SUBR  INIT;
          CHGVAR     &in30 '0'
          CHGVAR     &MsgS ' '
ENDSUBR;
```

Indicators

- Indicators are on/off switches used by programs. 2 possible values: ‘1’ or ‘0’
  - **Response indicators**: set by functions keys, used by programs to determine the appropriate USR response
    - e.g. exit when F3 is pressed
  - **Option indicators**: set by programs, used to control when/how info is displayed
    - e.g. an indicator is set to *on* when an error is detected causing an data field to be displayed in red.

Program Flow using Screens and Indicators



SNDRCVF

Check response indicators
DOWHILE (&IN03 *NE ‘1’)

Exit

Edit input, set option indictors
If (&name = ‘ ‘) chgvar IN30 ‘1’
SNDRCVF

Examples of Response Indicator Use

```
If  (&in03  *eq ‘1’) +
   goto  cmdlbl(exit)

If (&in05  * eq ‘1’)  +
  do
        chgvar  var(&assign1)  value ( 0 )
        chgvar  var(&assign2)  value ( 0 )
                goto  cmdlbl(read)
  enddo
```

Examples of Option Indicator Use

```
IF  (&ASSIGN1  *LT   0  *OR  &ASSIGN1   *GT   5)  +
        CHGVAR VAR( &IN30)  VALUE (‘1’)

IF  (&ASSIGN2   *LT  0  *OR  &ASSIGN2  *GT   10) +
        CHGVAR VAR (&IN31) VALUE (‘1’)

IF  (&IN30   *OR   &IN31) +
        GOTO  CMDLBL(SEND)    /* REDISPLAY SCREEN */
```

```
/* sample */
          DCLF     FILE(ORDERDF)
          SNDRCVF
          DOWHILE  (&IN03 *NE ‘1’)
                     IF  (&ORDERNO   *LE    0)   +
                          CHGVAR   VAR (&IN40 )   VALUE ( ‘1’ )
                    IF  (&ORDDESC   *EQ   ‘    ‘)  +
                          CHGVAR   VAR (&IN41 )   VALUE ( ‘1’)
                    IF  ( &IN40  *OR  &IN41 )  +
                         GOTO    CMDLBL(NEXT)
                    CALL  CBLPGM ( &ORDERNO  &ORDDESC )
    NEXT:           SNDRCVF
          ENDDO
           ENDPGM
```

Homework

- Summer 2020
  - You're not required to complete Lab 4-2 (i.e. Part E, F?, G)

The End