

This is the tutorial for IBM Debug Tool for z/OS<sup>®</sup>, one of the IBM zSeries<sup>®</sup> problem determination tools.

### Using Debug Tool's terminal interface

- Full-screen windows and navigation
- Using the debugger
  - Stepping through statements and running the program
  - Program statement breakpoints
  - Monitoring variables
  - Displaying variables in the log
  - Making breakpoints conditional
  - Variable change breakpoints
  - Program entry and exit breakpoints
  - Jumping to a statement
  - Ending the debugging session



This is the second of three sections that describes how to use the debugger.

This section will cover how to monitor variables, and how to display variables in the log.

## The SET AUTOMONITOR ON setting automatically displays variables referenced by the current statement

The screenshot shows the IBM Debug Tool interface. At the top, the command `SET AUTO ON` is entered at prompt `==`. A yellow box labeled `Enter` indicates the next action. The main window displays the AUTOMONITOR output:

```

***** AUTOMONITOR *****
0002 03 CUST-ID          01001
0003 02 RPT-CUST-ID     X'000000000'
*****
SOURCE: SAM1 +-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+-----8-----+
312         CALL 'SAM2' USING CUST-REC,
313           CUSTOMER-BALANCE-STATS
314         MOVE CUST-ID      TO RPT-CUST-ID
315         MOVE CUST-NAME    TO RPT-CUST-NAME
316         MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317         MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
LOG 0-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7-----+-----8-----+
0062 LIST AT ;
0063 There are no breakpoints set.
0064 SET AUTOMONITOR ON ;
PF 1:?          2:STEP          3:QUIT          4:LIST          5:FIND          6:
PF 7:UP         8:DOWN          9:GO           10:ZOOM        11:ZOOM LOG   12:
  
```

A callout box states: "Variables referenced by the current statement are displayed in the automonitor". In the screenshot, `CUST-ID` and `RPT-CUST-ID` in the MOVE statement are highlighted in red. A yellow box labeled `F2` is shown at the bottom right, and a `Step` button is also visible.

An important feature for watching variables is something called the auto monitor. The command: `SET AUTO ON` turns it on. The auto monitor automatically displays the values of variables referenced by the current statement. In this case, variables named `CUST-ID` and `RPT-CUST-ID` are referenced by the current statement, statement number 314. Those variables are shown automatically in the monitor window in the auto monitor area. Of course, when you step or go to another statement, a different set of variables will be shown.

F2 is pressed.

## After a **STEP**. The Automonitor displays variables referenced by the current statement.



```
COBOL LOCATION: SAM1 :> 315.1
Command ==>
Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 3
***** TOP OF MONITOR *****
-----1-----2-----3-----4-----
0001 ***** AUTOMONITOR *****
0002 02 CUST-NAME 'Lynn, Amanda'
0003 02 RPT-CUST-NAME
***** BOTTOM OF MONITOR *****

SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 312 OF 467
312 CALL 'SAM2' USING CUST-REC,
313 CUSTOMER-BALANCE-STATS
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE

LOG 0-----1-----2-----3-----4-----5----- LINE: 63 OF 65
0063 There are no breakpoints set.
0064 SET AUTOMONITOR ON ;
0065 STEP ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

4

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

That stepped to the next statement, 315, which is now the current statement. The auto monitor now shows the variables referenced by 315.

**The SET AUTOMONITOR ON BOTH setting displays variables from both the current and previous statements**

```

=====
==> SET AUTO ON BOTH
=====
Scroll ==> PAGE
R +---1---+---2---+---3---+---4---+---5---+---6- LINE: 1 OF 6
***** TOP OF MONITOR *****
-----1-----2-----3-----4-----
0001 ***** AUTOMONITOR SAM1 ::> SAM1 :> 315.1 *****
0002 02 CUST-NAME 'Lynn, Amanda' Current
0003 02 RPT-CUST-NAME 'Lynn, Amanda'
0004 ***** Previous Statement SAM1 ::> SAM1 :> 314.1 *****
0005 03 CUST-ID '01001' Previous
0006 02 RPT-CUST-ID '01001'
SOURCE: SAM1 +---1---+---2---+---3---+---4---+---5--- LINE: 312 OF 467
312 CALL 'SAM2' USING CUST-REC,
313 CUSTOMER-BALANCE-STATS
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
LOG 0 +---1---+---2---+---3---+---4---+---5---+---6- LINE: 64 OF 66
0064 SET AUTOMONITOR ON ;
0065 STEP ;
0066 SET AUTOMONITOR ON BOTH ;
PF 1: ? 2: STEP 3: QUIT 4: LIST 5: FIND 6: PAGE
PF 7: UP 8: DOWN 9: GO 10: ZOOM 11: ZOOM LOG 12: PAGE

```

5 IBM Debug Tool for z/OS V10 tutorial © 2012 IBM Corporation

By default, the auto monitor displays only variables referenced by the current statement. But if you specify the "both" option, in addition to displaying variables for the current statement, it also displays variables for the previously displayed statement. This is very helpful, because as you are stepping through a program, it will automatically show the **results** of each statement.

The command "SET AUTO ON BOTH" is entered, turning on the auto monitor with the "both" option. Notice that the current statement is 315, and that variables are displayed for both statements 315 and 314. Also notice that variable RPT-CUST-NAME will be changed when 315 runs.

F2 is pressed to step one statement.

After a STEP. The SET AUTO ON BOTH setting displays variables from the current and previous statements



```

COBOL  LOCATION: SAM1 :> 316.1
Command ==>
Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 6
***** TOP OF MONITOR *****
-----1-----2-----3-----4-----
0001  ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0002  02 CUST-OCCUPATION      'Musician
0003  02 RPT-CUST-OCCUPATION
0004  ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0005  02 CUST-NAME            'Lynn, Amanda
0006  02 RPT-CUST-NAME       'Lynn, Amanda
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 312 OF 467
312  CALL 'SAM2' USING CUST-REC,
313  CUSTOMER-BALANCE-STATS
314  MOVE CUST-ID            TO RPT-CUST-ID
315  MOVE CUST-NAME         TO RPT-CUST-NAME
316  MOVE CUST-OCCUPATION   TO RPT-CUST-OCCUPATION
317  MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
LOG 0-----1-----2-----3-----4-----5--- LINE: 65 OF 67
0065 STEP ;
0066 SET AUTOMONITOR ON BOTH ;
0067 STEP ;
PF 1:?          2:STEP      3:QUIT
PF 7:UP         8:DOWN      9:GO      10:ZOOM  11:ZOOM LOG 12:RETRIEVE

```

The BOTH automonitor option displays **results** automatically as you step through a program

Statement 315 ran, and 316 is now the current statement. You can see how RPT-CUST-NAME changed in the previous statement part of the auto monitor.

## A Find command searches for text in a window



```
LOCATION - SAM1 -> 316.1
==> F CUST-I FIRST $1.1
Scroll ==> PAGE
R +---1---+---2---+---3---+---4---+---5---+---6- LINE: 1 OF 6
***** TOP OF MONITOR *****
-----1-----2-----3-----4-----
0001 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0002 02 CUST-OCCUPATION 'Musician'
0003 02 RPT-CUST-OCCUPATION 'Musician'
0004 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0005 02 CUST-NAME 'Lynn, Amanda'
0006 02 RPT-CUST-NAME 'Lynn, Amanda'
SOURCE: SAM1 +---1---+---2---+---3---+---4---+---5---+---6- LINE: 65 OF 67
60 10 CUST-ID
61 10 CUST-RECORD-TYPE PIC X.
62 10 FILLER PIC X(7).
63 05 CUST-NAME PIC X(17).
64 05 CUST-ACCT-BALANCE PIC S9(7)V99 COMP-3.
65 05 CUST-ORDERS-YTD PIC S9(4) COMP.
LOG 0-----1-----2-----3-----4-----5-----6-----7----- LINE: 65 OF 67
0065 STEP ;
0066 SET AUTOMONITOR ON BOTH ;
0067 STEP ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:PAUSE
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:REPEAT
```

As you might expect, you can use find commands to find text strings. By default, the find command searches for text in the source window. The command: “F cust-i FIRST” is entered, and the first occurrence of that text string is located in the program source. The F5 key is repeat find. F5 is pressed several times.

## After repeated Finds



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR --+---1---+---2---+---3---+---4---+---5---+---6- LINE: 1 OF 6
***** TOP OF MONITOR *****
-----1-----2-----3-----4-----
0001 ***** AUTOMONITOR SAM1 :> SAM1 :> 316.1 *****
0002 02 CUST-OCCUPATION 'Musician'
0003 02 RPT-CUST-OCCUPATION '
0004 ***** Previous Statement SAM1 :> SAM1 :> 315.1 *****
0005 02 CUST-NAME 'Lynn, Amanda'
0006 02 RPT-CUST-NAME 'Lynn, Amanda'
SOURCE: SAM1 +---1---+---2---+---3---+---4---+---5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1---+---2---+---3---+---4---+---5--- LINE: 65 OF 67
0065 STEP ;
0066 SET AUTOMONITOR ON BOTH ;
0067 STEP ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

8

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

As you continue to press F5, the search continues forward in the program, and other occurrences of the string are found.



## FIND command



- **FIND** *string* , or
- **F** *string* , or
- **F** '*string*' , or
- **F** "*string*"
  - Finds the next occurrence of *string* in the source window
    - Or place your cursor in the log or monitor to search in that window
  - The default **F5** key finds the next occurrence
  - Frequently used command options:
    - **F string MON** searches in the monitor window
    - **F string LOG** searches in the log window
    - **F string PREV** searches upward from the current position
    - **F string FIRST** searches downward from the top
    - **F string LAST** searches upward from the bottom
    - **F string 10 20** searches only between columns 10 through 20, inclusive
    - **F string 1 20 FIRST LOG** shows that the above options can be combined



The syntax of a FIND command is F space, and then the text you are looking for. The string can be entered with or without surrounding apostrophes or quotation marks. If the string has embedded spaces, then surrounding apostrophes or quotation marks are needed.

Once you issue a find command, you can find the next occurrence with the F5 key.

You can also find text in the monitor or log windows. The MON keyword specifies that the search should be in the monitor window, and the LOG keyword specifies to search in the log.

The FIRST and LAST keywords make the search find the first or last occurrence of the string, and the search can be limited to specific column positions, by specifying the beginning and ending columns of the range.

## An M line command adds a monitor that displays all variables referenced by a statement



```

COBOL  LOCATION: SAM1 :> 316.1
Command ==>                                     Scroll ==> PAGE
MONITOR --+---1---+---2---+---3---+---4---+---5---+---6- LINE: 1 OF 6
***** TOP OF MONITOR *****
-----1-----2-----3-----4-----
0001  ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0002  02 CUST-OCCUPATION          'Musician
0003  02 RPT-CUST-OCCUPATION      '
0004  ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0005  02 CUST-NAME                'Lynn, Amanda
0006  02 RPT-CUST-NAME            'Lynn, Amanda
SOURCE: SAM1 +---1---+---2---+---3---+---4---+---5--- LINE: 314 OF 467
M 314  MOVE CUST-ID                TO RPT-CUST-ID
315  MOVE CUST-NAME              TO RPT-CUST-NAME
316  MOVE CUST-OCCUPATION        TO RPT-CUST-OCCUPATION
317  MOVE CUST-ACCT-BALANCE     TO RPT-CUST-ACCT-BALANCE
318  MOVE CUST-ORDERS-YTD       TO RPT-CUST-ORDERS-YTD
319  WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0---+---1---+---2---+---3---+---4---+---5---+--- LINE: 65 OF 67
0065  STEP ;
0066  SET AUTOMONITOR ON BOTH ;
0067  STEP ;
PF 1: ?      2: STEP      3: QUIT      4: LIST      5: FIND      6: PAUSE
PF 7: UP     8: DOWN     9: GO       10: ZOOM     11: ZOOM LOG  12: R

```



10

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

You can add specific variables to the monitor window, and there are several ways to do it. When you are working with programs generated by current compilers, such as Enterprise COBOL, Enterprise PL/I, and assembler, you can use M line commands in the source window to monitor variables.

Here, an M line command is entered on line 314. Notice that the statement references two variables, CUST-ID, and RPT-CUST-ID.

## Result of an M line command on statement 314



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
+-----1-----2-----3-----4-----
0001 1 CUST-ID '01001'
0002 RPT-CUST-ID '01001'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Amanda'
SOURCE: SAM1 +-----1-----2-----3-----4-----5- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0 +-----1-----2-----3-----4-----5- LINE: 67 OF 69
0067 STEP ;
0068 MONITOR
0069 LIST ( CUST-ID, RPT-CUST-ID ) ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

Monitor was added

The M line command added monitors for all the variables referenced by the statement. These monitors will remain in the monitor window as you continue to step and run the program. They are permanent, until you clear them. As the program runs, you can watch these variables change.

## An M<sub>n</sub> line command adds a monitor for the n<sup>th</sup> variable referenced by a statement



```

COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----1-----2-----3-----4-----
0001 1 CUST-ID '01001'
0002 RPT-CUST-ID '01001'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Amanda'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
M1 314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319
LOG 0-----5----- LINE: 67 OF 69
0067 STEP ;
0068 MONITOR
0069 LIST ( CUST-ID, RPT-CUST-ID ) ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:A
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:R

```

An M line command with a number (such as M1, M2, M3) monitors only one variable



You can optionally use an M line command to monitor only a single variable. For example, an M1 line command will add a monitor for only the first referenced variable. M2 will add a monitor for the second variable, and so on.

The line command M1 is entered on line 315.

# Result of an M1 line command on statement 315



```
COBOL LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 9
-----1-----2-----3-----4-----
0001 1 CUST-ID 01001
0002 RPT-CUST-ID 01001
0003 2 CUST-NAME 'Lynn, Amanda'
0004 ***** AUTOMONITOR SAM1 :> SAM1 :> 316.1 *****
0005 02 CUST-OCCUPATION 'Musician'
0006 02 RPT-CUST-OCCUPATION
0007 ***** Previous Statement SAM1 :> SAM1 :> 315.1
SOURCE: SAM1 +-----1-----2-----3-----4-----5-----
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 69 OF 71
0069 LIST ( CUST-ID, RPT-CUST-ID ) ;
0070 MONITOR
0071 LIST CUST-NAME ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

Monitor was added

Monitors remain active until you clear them

The M1 line command added a monitor for variable CUST-NAME, which is the first variable referenced by the statement.

## A C line command clears a monitor



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 9
-----1-----2-----3-----4-----
C 01 1 CUST-ID 01001
C 02 RPT-CUST-ID '01001'
C 03 2 CUST-NAME 'Lynn, Amanda'
0004 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0005 02 CUST-OCCUPATION 'Musician'
0006 02 RPT-CUST-OCCUPATION
0007 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 69 OF 71
0069 LIST ( CUST-ID, RPT-CUST-ID ) ;
0070 MONITOR
0071 LIST CUST-NAME ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:PAUSE
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:PAUSE
```

Enter

14

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

You can clear a monitor with a C line command in the monitor window. Here, C line commands are entered next to the two monitors that were just added.

## Result of C line commands - monitors were cleared



```

COBOL   LOCATION: SAM1 :> 316.1
Command ==> █                               Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 6
        -----1-----2-----3-----4-----
0001     ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0002     02 CUST-OCCUPATION           'Musician
0003     02 RPT-CUST-OCCUPATION       '
0004     ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0005     02 CUST-NAME                 'Lynn, Amanda
0006     02 RPT-CUST-NAME             'Lynn, Amanda
*****
***** BOTTOM OF MONITOR *****
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314     MOVE CUST-ID                 TO RPT-CUST-ID
315     MOVE CUST-NAME               TO RPT-CUST-NAME
316     MOVE CUST-OCCUPATION         TO RPT-CUST-OCCUPATION
317     MOVE CUST-ACCT-BALANCE      TO RPT-CUST-ACCT-BALANCE
318     MOVE CUST-ORDERS-YTD        TO RPT-CUST-ORDERS-YTD
319     WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0 +-----1-----2-----3-----4-----5-----+--- LINE: 71 OF 73
0071     LIST CUST-NAME ;
0072     CLEAR MONITOR 1 ;
0073     CLEAR MONITOR 2 ;
PF 1: ?      2: STEP      3: QUIT      4: LIST      5: FIND      6: AT/CLEAR
PF 7: UP     8: DOWN     9: GO       10: ZOOM     11: ZOOM LOG  12: RETRIEVE
    
```

15

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

The C line commands removed the monitors from the monitor window.

## A MONITOR LIST variable command adds a monitor for a variable



```

COBOL  LOCATION: SAM1 :> 316.1
Command ==> MONITOR LIST CUST-ID
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 6
-----1-----2-----3-----4-----
0001  ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0002  02 CUST-OCCUPATION          'Musician
0003  02 RPT-CUST-OCCUPATION
0004  ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0005  02 CUST-NAME                'Lynn, Amanda
0006  02 RPT-CUST-NAME            'Lynn, Amanda
*****
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314      MOVE CUST-ID              TO RPT-CUST-ID
315      MOVE CUST-NAME            TO RPT-CUST-NAME
316      MOVE CUST-OCCUPATION      TO RPT-CUST-OCCUPATION
317      MOVE CUST-ACCT-BALANCE   TO RPT-CUST-ACCT-BALANCE
318      MOVE CUST-ORDERS-YTD     TO RPT-CUST-ORDERS-YTD
319      WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0 +-----1-----2-----3-----4-----5--- LINE: 71 OF 73
0071  LIST CUST-NAME ;
0072  CLEAR MONITOR 1 ;
0073  CLEAR MONITOR 2 ;
PF 1: ?      2: STEP      3: QUIT      4: LIST      5: FIND      6: A
PF 7: UP     8: DOWN     9: GO       10: ZOOM     11: ZOOM LOG  12: R

```



16

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

M line commands can be used with current compilers, but are not available with programs that have been compiled with older out-of-support compilers such as OS/VS COBOL or VS COBOL II.

Another way to add a monitor is the "MONITOR LIST ..." command, and it can be used with any compiler. Here, the command "MONITOR LIST CUST-ID" is typed on the command line, and Enter is pressed.



## Result of a MONITOR LIST CUST-ID command



```
COBOL  LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +---1---+---2---+---3---+---4---+---5---+---6- LINE: 1 OF 7
0001  3 CUST-ID                                '01001'
0002  ***** AUTOMONITOR SAM1 :> SAM1 :> 316.1 *****
0003  02 CUST-OCCUPATION                        'Musician'
0004  02 RPT-CUST-OCCUPATION                    '
0005  ***** Previous Statement SAM1 :> SAM1 :> 315.1
0006  02 CUST-NAME                              'Lynn, Amanda'
0007  02 RPT-CUST-NAME                          'Lynn, Amanda'
SOURCE: SAM1 +---1---+---2---+---3---+---4---+---5--- LINE: 314 OF 467
314      MOVE CUST-ID                          TO RPT-CUST-ID
315      MOVE CUST-NAME                        TO RPT-CUST-NAME
316      MOVE CUST-OCCUPATION                  TO RPT-CUST-OCCUPATION
317      MOVE CUST-ACCT-BALANCE                TO RPT-CUST-ACCT-BALANCE
318      MOVE CUST-ORDERS-YTD                  TO RPT-CUST-ORDERS-YTD
319      WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0 +---1---+---2---+---3---+---4---+---5---+---6- LINE: 73 OF 75
0073  CLEAR MONITOR 2 ;
0074  MONITOR
0075  LIST CUST-ID ;
PF 1:?      2:STEP      3:QUIT      4:LIST      5:FIND      6:AT/CLEAR
PF 7:UP     8:DOWN     9:GO       10:ZOOM     11:ZOOM LOG  12:RETRIEVE
```

Monitor was added

Monitors remain active until you clear them

That added a monitor for the CUST-ID variable. A "MONITOR LIST ..." command has the same affect as an M line command, it is just another way to add a monitor for a variable.

## A MONITOR LIST command adds a monitor for a cursor-selected variable

```

COBOL  LOCATION: SAM1 :> 316.1
Command =<=> MON LIST                               Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 7
-----+-----1-----2-----3-----4-----
0001  3 CUST-ID                                '01001'
0002  ***** AUTOMONITOR SAM1 :> SAM1 :> 316.1 *****
0003  02 CUST-OCCUPATION                       'Musician'
0004  02 RPT-CUST-OCCUPATION
0005  ***** Previous Statement SAM1 :> SAM1 :> 315.1 *****
0006  02 CUST-NAME                             'Lynn, Amanda'
0007  02 RPT-CUST-NAME                         'Lynn, Amanda'
SOURCE: SAM1 +-----1-----2-----3-----4-----
314  MOVE CUST-ID TO RPT-CUST-ID
315  MOVE CUST-NAME TO RPT-CUST-NAME
316  MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317  MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318  MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319  WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5-----6- LINE: 73 OF 75
0073  CLEAR MONITOR 2 ;
0074  MONITOR
0075  LIST CUST-ID ;
PF 1:?          2:STEP          3:QUIT          4:LIST          5:FIND          6:A
PF 7:UP         8:DOWN          9:GO           10:ZOOM         11:ZOOM LOG    12:R

```

Save typing by cursor-selecting the variable to be monitored



Here are some keystroke-saving tips. First, any keyword in a command can be abbreviated. You can type the least number of letters necessary to differentiate each keyword from other keywords. So for example, instead of typing "MONITOR LIST", try just "MON LIS".

A tip just for the monitor list command is that you do not have to type in the variable name. Type the command "MONITOR LIST" on the command line, or an abbreviation, as shown. But do not press Enter yet. Place the cursor on a variable in the source window that you want to monitor, and then press Enter.

## Result of a MONITOR LIST command

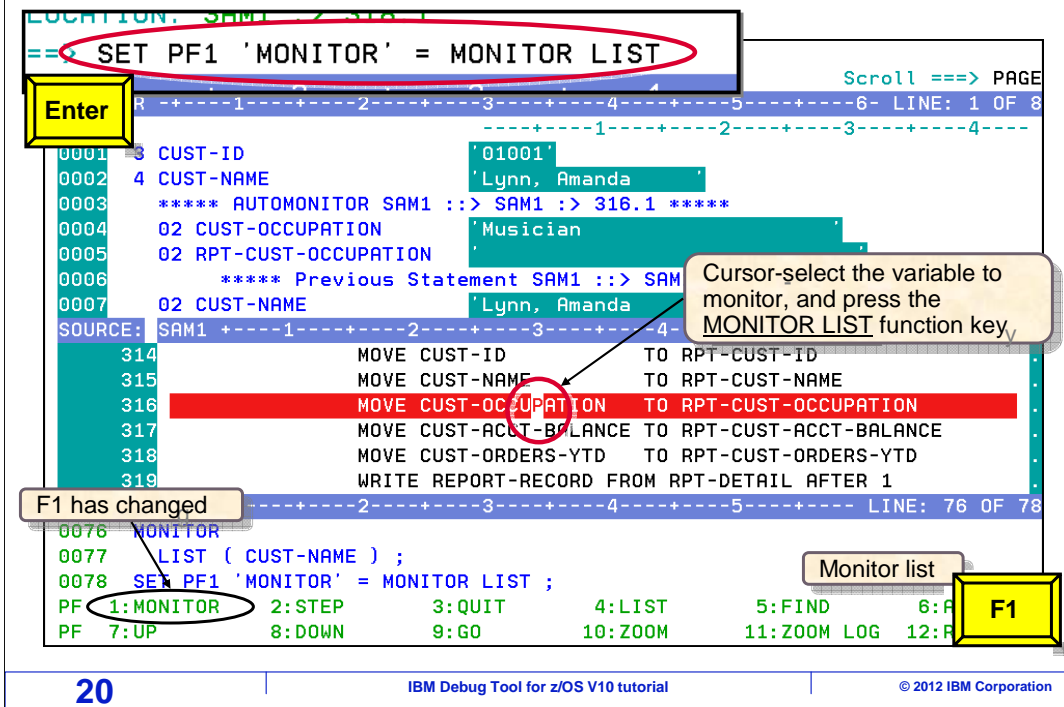


```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----1-----2-----3-----4-----
0001 3 CUST-ID '01001'
0002 4 CUST-NAME 'Lynn, Amanda'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Amanda'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 75 OF 77
0075 LIST CUST-ID ;
0076 MONITOR
0077 LIST ( CUST-NAME ) ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

Monitor was added

That added a monitor for the selected variable.

## Tip: Set up a function key for the MONITOR LIST command



```
LOCATION: SAM1 7 STEP 1
==> SET PF1 'MONITOR' = MONITOR LIST
Scroll ==> PAGE
R +---1---+---2---+---3---+---4---+---5---+---6- LINE: 1 OF 8
-----1-----2-----3-----4-----
0001 3 CUST-ID          01001
0002 4 CUST-NAME      'Lynn, Amanda'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 ::> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM
0007 02 CUST-NAME      'Lynn, Amanda'
SOURCE: SAM1 +---1---+---2---+---3---+---4-
314 MOVE CUST-ID          TO RPT-CUST-ID
315 MOVE CUST-NAME      TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
-----2-----3-----4-----5-----6----- LINE: 76 OF 78
0076 MONITOR
0077 LIST ( CUST-NAME ) ;
0078 SET PF1 'MONITOR' = MONITOR LIST ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:R
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:R
```

You may prefer to have a function key defined for the "monitor list" command. If you do, it can be just that much quicker to monitor variables. In this example, a command was entered to change the setting of the F1 key to the "monitor list" command. In the function key display at the bottom of the screen, notice the F1 key setting has been changed.

Once you have a function key set like this, it can be used to add a monitor. In this example, the cursor is placed on a variable, and the F1 function key is pressed.

## Result of using a customized MONITOR LIST key



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 9
-----+-----1-----2-----3-----4-----
0001 3 CUST-ID '01001'
0002 4 CUST-NAME 'Lynn, Amanda'
0003 5 CUST-OCCUPATION 'Musician'
0004 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0005 02 CUST-OCCUPATION 'Musician'
0006 02 RPT-CUST-OCCUPATION
0007 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----+-----1-----2-----3-----4-----5----- LINE: 78 OF 80
0078 SET PF1 'MONITOR' = MONITOR LIST ;
0079 MONITOR
0080 LIST ( CUST-OCCUPATION ) ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

Monitor was added

That added a monitor for the selected variable.

## A C line command clears a monitor



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 9
-----+-----1-----2-----3-----4-----
C 0001 3 CUST-ID 01001
0002 4 CUST-NAME 'Lynn, Amanda
0003 5 CUST-OCCUPATION 'Musician
0004 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0005 02 CUST-OCCUPATION 'Musician
0006 02 RPT-CUST-OCCUPATION
0007 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----+-----1-----2-----3-----4-----5----- LINE: 78 OF 80
0078 SET PF1 'MONITOR' = MONITOR LIST ;
0079 MONITOR
0080 LIST ( CUST-OCCUPATION ) ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:PAUSE
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:PAGE
```



Remember that a C line command in the monitor window clears a monitor. "C" is entered next to a monitor.

## Result of a C line command - a monitor was cleared



```

COBOL  LOCATION: SAM1 :> 316.1
Command ==>
Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----1-----2-----3-----4-----
0001  4 CUST-NAME           'Lynn, Amanda'
0002  5 CUST-OCCUPATION    'Musician'
0003  ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004  02 CUST-OCCUPATION   'Musician'
0005  02 RPT-CUST-OCCUPATION
0006  ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007  02 CUST-NAME         'Lynn, Amanda'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314      MOVE CUST-ID           TO RPT-CUST-ID
315      MOVE CUST-NAME        TO RPT-CUST-NAME
316      MOVE CUST-OCCUPATION   TO RPT-CUST-OCCUPATION
317      MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318      MOVE CUST-ORDERS-YTD   TO RPT-CUST-ORDERS-YTD
319      WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 79 OF 81
0079  MONITOR
0080  LIST ( CUST-OCCUPATION ) ;
0081  CLEAR MONITOR 3 ;
PF 1:MONITOR  2:STEP      3:QUIT    4:LIST    5:FIND    6:AT/CLEAR
PF 7:UP       8:DOWN     9:GO     10:ZOOM   11:ZOOM LOG 12:RETRIEVE
    
```

Tip: a pair of CC line commands will remove a block of variables

And the monitor was cleared. When a variable is displayed in the monitor window, you can change its value. Notice the current value of the CUST-NAME field: "Lynn, Amanda".

## Overtyping the value of a monitored item to change it



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----+-----1-----2-----3-----4-----
0001 4 CUST-NAME          Lynn, Viola
0002 5 CUST-OCCUPATION   'Musician'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION   'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME        'Lynn, Amanda'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID          TO RPT-CUST-ID
315 MOVE CUST-NAME       TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----+-----1-----2-----3-----4-----5----- LINE: 79 OF 81
0079 MONITOR
0080 LIST ( CUST-OCCUPATION ) ;
0081 CLEAR MONITOR 3 ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:A
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:R
```



The value is overtyped to a different value, and Enter is pressed.



## The value of a variable was changed by overtyping it in the monitor



```
COBOL LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----1-----2-----3-----4-----
0001 4 CUST-NAME 'Lynn, Viola'
0002 5 CUST-OCCUPATION 'Musician'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Viola'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 80 OF 82
0080 LIST ( CUST-OCCUPATION ) ;
0081 CLEAR MONITOR 3 ;
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

25

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

That changed the value of the variable. As the program continues to run, it will see the updated value.

## Use an H (hex format) line command in the monitor to display an item in hex format



```

COBOL LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----1-----2-----3-----4-----
H 0001 4 CUST-NAME          'Lynn, Viola'
0002 5 CUST-OCCUPATION   'Musician'
0003 ***** AUTOMONITOR SAM1 :> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION   'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 :> SAM1 :> 315.1 *****
0007 02 CUST-NAME          'Lynn, Viola'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID          TO RPT-CUST-ID
315 MOVE CUST-NAME       TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 80 OF 82
0080 LIST ( CUST-OCCUPATION ) ;
0081 CLEAR MONITOR 3 ;
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
PF 1: ?      2: STEP      3: QUIT      4: LIST      5: FIND      6: A
PF 7: UP     8: DOWN     9: GO       10: ZOOM     11: ZOOM LOG  12: R

```



Variables can be displayed in the monitor window in two formats: the default format, and in hexadecimal format. In the default format, character values are displayed as text, and the decimal values of numeric fields are displayed.

You can use an "H" line command to display a monitor in hexadecimal format. An "H" line command is typed next to a monitor, and Enter is pressed.

## Result of H (hex format) line command in the monitor window



```
COBOL LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
+-----1-----2-----3-----4-----
0001 4 CUST-NAME X'D3A895956B40E5899693814040404040'
0002 5 CUST-OCCUPATION 'Musician'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Viola'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 81 OF 83
0081 CLEAR MONITOR 3 ;
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
0083 MONITOR 4 HEX ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

27

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

Now the selected monitor is displayed in hexadecimal format, instead of in the default format.

## Use an D (default format) line command in the monitor to reset an item to its default display format



```

COBOL LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
-----1-----2-----3-----4-----
D 0001 4 CUST-NAME X'D3A895956B40E5899693814040404040'
0002 5 CUST-OCCUPATION 'Musician'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Viola'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 81 OF 83
0081 CLEAR MONITOR 3 ;
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
0083 MONITOR 4 HEX ;
PF 1: ? 2: STEP 3: QUIT 4: LIST 5: FIND 6: A
PF 7: UP 8: DOWN 9: GO 10: ZOOM 11: ZOOM LOG 12: R

```



You can change the display back to default format with a "D" line command. A "D" is entered next to the monitor, and Enter is pressed.

## Result of D (default format) line command in the monitor window



```
COBOL LOCATION: SAM1 :> 316.1
Command ==>
MONITOR +-----1-----2-----3-----4-----5-----6- LINE: 1 OF 8
0001 4 CUST-NAME 'Lynn, Viola'
0002 5 CUST-OCCUPATION 'Musician'
0003 ***** AUTOMONITOR SAM1 ::> SAM1 :> 316.1 *****
0004 02 CUST-OCCUPATION 'Musician'
0005 02 RPT-CUST-OCCUPATION
0006 ***** Previous Statement SAM1 ::> SAM1 :> 315.1 *****
0007 02 CUST-NAME 'Lynn, Viola'
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 82 OF 84
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
0083 MONITOR 4 HEX ;
0084 MONITOR 4 DEFAULT ;
PF 1:? 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

29

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

That changed the display of the monitor back to its default format.

## Monitoring group variables



```
LOCATION - SEMI - 7 318.1
==> MON LIS CUST-REC
Enter
0001 04 CUST-NAME          Lynn, Viola
0002 05 CUST-OCCUPATION   Musician
0003 06 01 CUST-REC
0004 02 CUST-KEY
0005 03 CUST-ID           '01001'
0006 03 CUST-RECORD-TYPE 'C'
0007 03 FILLER
SOURCE SAM1 +-----1-----2-----+-----3-----4-----+-----5-----6 LINE: 1 OF 18
314 MOVE CUST-ID
315 MOVE CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----+-----3-----4-----+-----5-----+-----6 LINE: 82 OF 84
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
0083 MONITOR
0084 LIST CUST-REC ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

When a group-level variable is monitored, its subordinate variables are also displayed

30 IBM Debug Tool for z/OS V10 tutorial © 2012 IBM Corporation

So far, all of the variables monitored have been simple variables, not groups or structures. Here, a group level variable is added to the monitor. Be aware that when you monitor a group variable, such as an 01 level in COBOL or a structure in PL/I, all of its subordinate variables are also displayed.

## A MONITOR LIST TITLED WSS command adds a monitor containing all working-storage variables



```

COBOL LOCATION: SAM1 -> 316.1
Command ==> MON LIS TI WS; Z MON
MONITOR --+-----1-----2-----3-----4-----5-----6 LINE: 1 OF 18
-----+-----1-----2-----3-----4-----
0001  4 CUST-NAME          'Lynn, Viola
0002  5 CUST-OCCUPATION   Musician
0003  6 01 CUST-REC
0004  02 CUST-KEY
0005  03 CUST-ID          '01001'
0006  03 CUST-RECORD-TYPE 'C'
0007  03 FILLER
SOURCE: SAM1 +-----1-----2-----3-----4-----
314          MOVE CUST-ID          TO RPT-CUST-ID
315          MOVE CUST-NAME        TO RPT-CUST-NAME
316          MOVE CUST-OCCUPATION   TO RPT-CUST-OCCUPATION
317          MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318          MOVE CUST-ORDERS-YTD   TO RPT-CUST-ORDERS-YTD
319          WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----+-----1-----2-----3-----4-----5-----6 LINE: 82 OF 84
0082 MOVE 'Lynn, Viola' TO CUST-NAME ;
0083 MONITOR
0084 LIST CUST-REC ;
PF 1:MONITOR  2:STEP      3:QUIT    4:LIST    5:FIND    6:PAUSE
PF 7:UP       8:DOWN     9:GO      10:ZOOM   11:ZOOM LOG 12:RECALL

```

Tip: Keywords in commands can be abbreviated to the fewest characters that differentiate them from other keywords

Z MON = ZOOM MONITOR



Another important command, when working with COBOL programs, is "monitor list titled wss". In this example, it is abbreviated. The command "zoom monitor" is also entered, to expand the monitor window to full screen. Enter is pressed.

## Result of MONITOR LIST TITLED WSS command All working-storage variables are displayed



```

COBOL  LOCATION: SAM1 :> 316.1
Command ==> F STATS                               Scroll ==> PAGE
MONITOR --+-----1-----2-----3-----4-----5-----6-----7-----8-----9-----10-----11-----12----- LINE: 1 OF 168
-----+-----1-----2-----3-----4-----
0001  4 CUST-NAME                               'Lynn, Viola'
0002  5 CUST-OCCUPATION                          'Musician'
0003  6 01 CUST-REC
0004  02 CUST-KEY
0005  03 CUST-ID                               '01001'
0006  03 CUST-RECORD-TYPE                       'C'
0007  03 FILLER
0008  02 CUST-NAME                               'Lynn, Viola'
0009  02 CUST-ACCT-BALANCE                       +0000067.68
0010  02 CUST-ORDERS-YTD                         +00009
0011  02 CUST-CITY                              'Spirit Lake'
0012  02 CUST-OCCUPATION                          'Musician'
0013  7 01 SYSTEM-DATE-AND-TIME
0014  02 CURRENT-DATE
0015  03 CURRENT-YEAR                           09
0016  03 CURRENT-MONTH                         12
0017  03 CURRENT-DAY                           23
0018  02 CURRENT-TIME
PF 1: MONITOR  2: STEP  3: QUIT  4: LIST  5: FIND  6: A
PF 7: UP       8: DOWN  9: GO   10: ZOOM 11: ZOOM LOG 12: R

```

The new monitor (working storage) starts here

Enter

32 IBM Debug Tool for z/OS V10 tutorial © 2012 IBM Corporation

The monitor window is zoomed in. The "monitor list titled wss" command displayed all of the variables in the program's working storage area in the monitor window.

With so many variables displayed, a "find" command can be very useful to locate variables in the monitor. The command "find stats" is entered.



## Result of Find command



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR --+---1---+---2---+---3---+---4---+---5---+--- LINE: 52 OF 168
-----+---1---+---2---+---3---+---4---
0052 01 CUSTOMER-BALANCE-SSTATS
0053 02 BALANCE-COUNT +0000001.00
0054 02 BALANCE-TOT +0000067.68
0055 02 BALANCE-MIN +0000067.68
0056 02 BALANCE-MAX +0000067.68
0057 02 BALANCE-RANGE +0000000.00
0058 02 BALANCE-AVG +0000067.68
0059 01 PRODUCT-STATS
0060 02 SERV-CALLS-COUNT +0000000
0061 02 SERV-CALLS-TOTAL +0000000
0062 02 SERV-CALLS-MIN +0000000
0063 02 SERV-CALLS-MAX +0000000
0064 02 SERV-CALLS-RANGE +0000000
0065 02 SERV-CALLS-AVG +0000000.00
0066 01 RPT-HEADER1
0067 02 FILLER 'CUSTOMER FILE REPORT DATE: '
0068 02 RPT-MM 12
0069 02 FILLER '/'
PF 1: MONITOR 2: STEP 3: QUIT 4: LIST 5: FIND 6: AT/CLEAR
PF 7: UP 8: DOWN 9: GO 10: ZOOM 11: ZOOM LOG 12: RETRIEVE
```

33

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

That found a variable with the string "stats" in the name.

## Commands to monitor variables AUTOMONITOR



- **SET AUTOMONITOR ON** , or
- **SET AUTO ON**
  - Adds a monitor that will automatically display variables referenced by the current statement
- **SET AUTO ON BOTH**
  - Adds a monitor that will automatically display variables referenced by *both* the current statement, and the previously displayed statement
  - Shows the *results* of statements as you step
- Frequently used command options:
  - **SET AUTO OFF** clears the automonitor
  - **SET AUTO ON BOTH LOG** displays the automonitor, and writes the same information to the log for *every* statement executed (use sparingly)
- Automonitor is not available with some older, out of support compilers

34

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

Here is a review of commands you can use to monitor variables.

"Set auto on" turns on the auto monitor, which will automatically display variables referenced by the current statement. It is typical to have the auto monitor turned on while you are debugging.

"Set auto on both" turns on the auto monitor, and in addition to variables referenced by the current statement, also displays variables referenced by the previously displayed statement. It shows the results of statements as you step through a program.

If you want to turn off the auto monitor, the command "set auto off" will clear it from the monitor window.

Another option that can be helpful is the "log" option. For example, the command "set auto on both log" not only turns on the auto monitor, it also writes the same information to the log window for every statement that runs. This can give you a complete documented history of the results of every statement. But use it sparingly, since it produces a lot of output in the log. It is possible to have a "log file", and if you do, it would receive all of that information.

The auto monitor is not available with certain older, out-of support compilers, such as OS/VS COBOL and VS COBOL II.

## Commands to monitor variables

### M (monitor) line commands



- **M** line command in the source window
  - Adds a monitor (an entry in the monitor window) that displays all variables referenced by the statement
- **M<sub>n</sub>** line command in the source window
  - Adds a monitor that displays the n<sup>th</sup> variable referenced by the statement
- M and M<sub>n</sub> line commands are not available with some older, out of support compilers

35

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

You can use an M line command in the source window to add a monitor to display all the variables referenced by a statement. If you only want to monitor one variable, enter an M line command with a number, to indicate that you want to monitor only the nth referenced variable.

M line commands are not available when debugging programs compiled with some older, out-of support compilers.

## Commands to monitor variables MONITOR LIST



- **MONITOR LIST** *variable-name* , or
- **MONITOR LIST** (with the cursor placed on a variable)
  - Adds a monitor (an entry in the monitor window) that displays the variable
  - Consider defining a function key for MONITOR LIST
    - For example, to set the F1 key:
      - SET PF1 'MONITOR' = MONITOR LIST
    - The suggested setting for a "monitor list" function key is:
      - SET PF1 'MONITOR' = MONITOR LOCAL %CU LIST
        - The 'LOCAL %CU' setting is recommended. It hides the monitor entry when running in a different program.
- **C** line command in the monitor window
  - Clears a monitor
- **CLEAR MONITOR**
  - Clears all entries from the monitor window

36

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

You can also use a "monitor list variable-name" command to add a monitor for a variable. Remember that you can abbreviate commands, so it can be shortened to just "mon lis". You can easily monitor a variable by typing "monitor list" on the command line, and then placing your cursor on the variable you want to monitor, before pressing Enter.

Also, consider defining a function key for the monitor list command. If you do, consider using the second example, which has the "local %cu" option, which will hide the monitor when you are running in different programs, such as subroutines.

A "C" line command clears a monitor, and you can clear everything that is in the monitor window with the single command: "clear monitor".

## Commands to change how monitored items are displayed

- **H** line command in the monitor window
  - Displays the selected item in hexadecimal format
- **D** line command in the monitor window
  - Displays the selected item in the default format for it's declared data type
    - Text for character variables, decimal values for numeric variables
    - For some programming languages, variables containing bad data are automatically displayed in hexadecimal format by default

There are line commands in the monitor window you can use to change the display format. An "H" line command changes the display of a monitor to hexadecimal. And a "D" line command changes the display back to the default format, showing text for character variables, and decimal values for numeric variables.

## Commands to monitor variables MONITOR LIST TITLED



- **MONITOR LIST TITLED WSS** , or (note 1)
- **MON LIST TI WS** (note 1)
  - Adds a monitor that displays all variables in the working-storage section
- **MON LIST TI FS** (note 1)
  - Adds a monitor that displays all variables in the file section
- **MON LIST TI LS** (note 1)
  - Adds a monitor that displays all variables in the linkage section
- **MON LIST TI \*** (note 2)
  - Adds a monitor that displays all variables in the current program

Note 1: COBOL only

Note 2: COBOL, C, and C++ only

There are other important options for working with COBOL, C, and C++ programs. A "monitor list titled wss" command adds a monitor that displays all of the variables in the working-storage section of your COBOL program.

Use "monitor list titled fs" or "monitor list titled ls" to display all variables from the file section or linkage section of your COBOL programs.

The command "monitor list titled \*" will add a monitor that displays all of the variables in your COBOL, C, or C++ program.

### Using Debug Tool's terminal interface

- Full-screen windows and navigation
- Using the debugger
  - Stepping through statements and running the program
  - Program statement breakpoints
  - Monitoring variables
  - Displaying variables in the log
  - Making breakpoints conditional
  - Variable change breakpoints
  - Program entry and exit breakpoints
  - Jumping to a statement
  - Ending the debugging session



Next, you will see how to display variables in the log.

## An L line command lists all variables referenced by a statement



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5----- LINE: 52 OF 168
-----+-----1-----2-----3-----4-----
0052 01 CUSTOMER-BALANCE-STATS
0053 02 BALANCE-COUNT +0000001.00
0054 02 BALANCE-TOT +0000067.68
0055 02 BALANCE-MIN +0000067.68
0056 02 BALANCE-MAX +0000067.68
0057 02 BALANCE-RANGE +0000000.00
0058 02 BALANCE-AVG +0000067.68
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
L 314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 84 OF 86
0084 LIST CUST-REC ;
0085 MONITOR
0086 LIST TITLED WSS ;
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:PAUSE
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:PRINT
```

Enter

40 IBM Debug Tool for z/OS V10 tutorial © 2012 IBM Corporation

Earlier, you saw that you can use "M" line commands in the source window to display variables in the monitor window. You can use an "L" line command to list variables in the log window instead. Here, an "L" line command is entered next to line 315.



## Result of an L line command on statement 315



```

COBOL   LOCATION: SAM1 :> 316.1
Command ==>
Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5----- LINE: 52 OF 168
-----1-----2-----3-----4-----
0052    01 CUSTOMER-BALANCE-STATS
0053    02 BALANCE-COUNT              +0000001.00
0054    02 BALANCE-TOT                +0000067.68
0055    02 BALANCE-MIN                +0000067.68
0056    02 BALANCE-MAX                +0000067.68
0057    02 BALANCE-RANGE              +0000000.00
0058    02 BALANCE-AVG                +0000067.68
SOURCE: SAM1 +-----1-----2-----3-----4-----5--- LINE: 314 OF 467
314      MOVE CUST-ID                 TO RPT-CUST-ID
315      MOVE CUST-NAME                TO RPT-CUST-NAME
316      MOVE CUST-OCCUPATION          TO RPT-CUST-OCCUPATION
317      MOVE CUST-ACCT-BALANCE        TO RPT-CUST-ACCT-BALANCE
318      MOVE CUST-ORDERS-YTD          TO RPT-CUST-ORDERS-YTD
319      WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 87 OF 89
0087    LIST ( CUST-NAME, RPT-CUST-NAME ) ;
0088    CUST-NAME = 'Lynn, Viola
0089    RPT-CUST-NAME = 'Lynn, Amanda
PF  4:MONITOR  2:STEP  3:QUIT  4:LIST  5:FIND  6:AT/CLEAR
PF  7:UP      8:DOWN  9:GO   10:ZOOM 11:ZOOM LOG 12:RETRIEVE
    
```

An L line command with a number (such as L1, L2, L3) lists only one variable

That listed the values of all the variables referenced by the statement in the log.

## Use the LIST function key (default=F4) to list a cursor-selected variable



```

COBOL  LOCATION: SAM1 :> 316.1
Command ==>                               Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5----- LINE: 52 OF 168
-----+-----1-----2-----3-----4-----
0052  01 CUSTOMER-BALANCE-STATS
0053  02 BALANCE-COUNT                +0000001.00
0054  02 BALANCE-TOT                  +0000067.68
0055  02 BALANCE-MIN                  +0000067.68
0056  02 BALANCE-MAX                  +0000067.68
0057  02 BALANCE-RANGE                +0000000.00
0058  02 BALANCE-AVG                  +0000067.68
SOURCE: SAM1 +-----1-----2-----3-----4-----5----- LINE: 87 OF 89
314      MOVE CUST-ID
315      MOVE CUST-NAME                TO RPT-CUST-NAME
316      MOVE CUST-OCCUPATION          TO RPT-CUST-OCCUPATION
317      MOVE CUST-ACCT-BALANCE        TO RPT-CUST-ACCT-BALANCE
318      MOVE CUST-ORDERS-YTD          TO RPT-CUST-ORDERS-YTD
319      WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----+-----1-----2-----3-----4-----5----- LINE: 87 OF 89
0087  LIST ( CUST-NAME, RPT-CUST-NAME ) ;
0088  CUST-NAME = 'Lynn, Viola
0089  RPT-CUST-NAME = 'Lynn, Amanda
PF 1:MONITOR  2:STEP    3:QUIT    4:LIST    5:FIND    6:PAUSE
PF 7:UP       8:DOWN    9:GO      10:ZOOM   11:ZOOM LOG 12:RECALL
    
```

Quickly list a variable by cursor-selecting it and pressing the LIST function key



Another way to list a variable in the log is with a "list" command. Notice that by default the F4 key is set to the "list" command. In this example, the cursor is placed on a variable in the source window, and F4 is pressed.

## Result of using the LIST function key and a cursor-selected variable



```
COBOL LOCATION: SAM1 :> 316.1
Command ==> Scroll ==> PAGE
MONITOR +-----1-----2-----3-----4-----5----- LINE: 52 OF 168
-----1-----2-----3-----4-----
0052 01 CUSTOMER-BALANCE-STATS
0053 02 BALANCE-COUNT +0000001.00
0054 02 BALANCE-TOT +0000067.68
0055 02 BALANCE-MIN +0000067.68
0056 02 BALANCE-MAX +0000067.68
0057 02 BALANCE-RANGE +0000000.00
0058 02 BALANCE-AVG +0000067.68
SOURCE: SAM1 +-----1-----2-----3-----4-----5----- LINE: 314 OF 467
314 MOVE CUST-ID TO RPT-CUST-ID
315 MOVE CUST-NAME TO RPT-CUST-NAME
316 MOVE CUST-OCCUPATION TO RPT-CUST-OCCUPATION
317 MOVE CUST-ACCT-BALANCE TO RPT-CUST-ACCT-BALANCE
318 MOVE CUST-ORDERS-YTD TO RPT-CUST-ORDERS-YTD
319 WRITE REPORT-RECORD FROM RPT-DETAIL AFTER 1
LOG 0-----1-----2-----3-----4-----5----- LINE: 89 OF 91
0089 RPT-CUST-NAME = 'Lynn, Amanda
0090 LIST ( CUST-OCCUPATION ) ;
0091 CUST-OCCUPATION = 'Musician
PF 1:MONITOR 2:STEP 3:QUIT 4:LIST 5:FIND 6:AT/CLEAR
PF 7:UP 8:DOWN 9:GO 10:ZOOM 11:ZOOM LOG 12:RETRIEVE
```

43

IBM Debug Tool for z/OS V10 tutorial

© 2012 IBM Corporation

The variable is displayed in the log. That is a quick way to display the value of a variable.

Now you have seen two ways to see a variable – you can add a monitor for it in the monitor window, or you can list it in the log. When would you use one method over the other? Generally, you may want to only monitor only certain variables, so that you can always see the current values of the most important variables in the monitor. If you only need to see the value of a variable once, and you do not need to watch it continuously, consider listing it in the log. That way, your monitor does not get filled up with things you do not need. Another difference is that a log entry does not change. If you list a variable in the log, you can review it later and you have a history of the value the variable at the time you listed it.

Use any combination of monitor and list that you prefer to display your variables.

## Commands to list variables in the log



- L line command in the source window
  - Displays all variables referenced by the statement
- Ln line command in the source window
  - Displays the n<sup>th</sup> variable referenced by the statement
- L and Ln line commands are not available with some older, out of support compilers
  
- LIST *variable-name* , or
- LIST (with the cursor placed on a variable), or
- The default F4 key (with the cursor placed on a variable)
  - Displays the variable

You can use an "L" line command in the source window to list all of the variables referenced by a statement. If you only want to list one variable, enter an L line command with a number, to indicate that you want to list only the nth referenced variable.

"L" line commands are not available when working with programs compiled with some older, out-of support compilers.

A "list variable-name" command can be used instead of an "L" line command. You can also list a variable by typing "list" on the command line, and then placing your cursor on the variable you want to monitor, before pressing Enter. Also, by default, the F4 key is set to the "list" command. So you can quickly list a variable just by placing your cursor on the variable you want to list, and pressing the F4 key.

That is the end the second of three sections that describes how to use the debugger. Continue with the third part.

## Feedback



Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send email feedback:

[mailto:iea@us.ibm.com?subject=Feedback\\_about\\_DTv12\\_TutorialTemplate.ppt](mailto:iea@us.ibm.com?subject=Feedback_about_DTv12_TutorialTemplate.ppt)

This module is also available in PDF format at: [../DTv12\\_TutorialTemplate.pdf](#)

You can help improve the quality of IBM Education Assistant content by providing feedback.



## Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, z/OS, and zSeries are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2012. All rights reserved.