


[print](#) | [close](#)

APIs by Example: Listing and Working with Validation List Entries

[System iNetwork Programming Tips Newsletter](#)

[Carsten Flensburg](#)

Carsten Flensburg

Thu, 08/27/2009 (All day)

Today I'm offering you the last piece of the Validation List puzzle I've presented in recent installments of the APIs by Example series. The pieces provided so far include the WRKVLDL, CLRVLDL, CVTVLDL, ADDVLDLE, DSPVLDLE, VFYVLDLE, CHGVLDLE, and RMVVLDLE CL commands. To complete the series, I provide the Work with Validation List Entries (WRKVLDLE) command as well as the Validation List Menu (CMDVLDL2) that consolidates all the validation list utilities I've presented.

I've also added a minor enhancement to the previously published DSPVLDLE utility to include the validation list entry's encryption attribute. Furthermore, I've updated the command's parameter interface to make it consistent with the other validation list entry commands. When you total them up, you'll notice that there are quite a lot of source members to download at the end of this article!

The Work with Validation List Entry (WRKVLDLE) command is the missing link in the catalogue of validation list commands presented so far. Following the compilation of the sources included with this article a new option 12 on the Work with Validation Lists list panel will surface:

Work with Validation Lists

WYNDHAMW
22-08-09

21:22:11

List order . . . : *VLDL
Position to . . .

Type options, press Enter.

2=Edit authority
object

3=Copy

4=Delete

7=Rename

8=Work with

12=Work with entries

13=Change description

14=Clear

Validation		Entry	
Opt	List	Library	Count Text
	QGLDVLDL	QUSRSYS	4
	QSASRVID2B	QUSRSYS	7

QSJSRVID	QUSRSYS	1
QTOCPTP	QUSRSYS	5
QTOCSASVL	QUSRSYS	7
QTOVDVPKEY	QUSRSYS	2
QTOVDVSKEY	QUSRSYS	0
QTOZRADI	QUSRSYS	0

Bottom
Parameters or command

===>

F3=Exit F4=Prompt F5=Refresh F6=Create validation list
F9=Retrieve
F11=Display full text F12=Cancel F17=Top F18=Bottom

Taking option 12 for one of the displayed validation lists (or running the WRKVLDLE command directly from a command line) returns a panel showing the validation list entries and offering a number of options (Verify, Change, Remove, Display, Print) as well as command keys. These options call the validation list entry commands that I made available in previous articles in this article series. Here's what that list looks like if the WRKVLDLE command is run on my system, specifying the QSASRVID2B validation list in library QUSRSYS as the target:

```

                                Work with Validation List Entries

WYNDHAMW                                                                22-08-09
21:22:42
Validation list   :   QSASRVID2B

Library   . . . :   QUSRSYS                Position to   . . .

Type options, press Enter.

1=Verify   2=Change   4=Remove   5=Display   6=Print

Opt  Entry ID                                Len  CCSID  Entry Data

```

```

0          9      37  940665-42CAC
V5R2M0P20003IQ2004-0 >
1          9      37  940665-D9C52  V5R2M0
  >
2          9      37  940665-42CAC
V5R2M0P20003IQ2004-0 >
3          9      37  940665-42CAC
V5R2M0P20003IQ2004-0 >
4          9      37  940665-D9C52  V5R2M0
  >
5          9      37  940665-D9C52  V5R2M0
  >
6          9      37  940665-42CAC
V5R2M0P20003IQ2004-0 >

Bottom
Parameters or command

===>

F3=Exit    F4=Prompt    F5=Refresh    F6=Add entry    F9=Retrieve
F12=Cancel
F17=Top     F18=Bottom    F22=Display entire field value    F24=More
keys

```

If you're looking for a particular entry ID in a large list, you can use the *Position to* function to position the list to that entry. Note that the list is always sorted in the order dictated by each validation list entry ID's binary value. Since the binary value of the same letter in lower case versus upper case is often far apart, it is quite important that you consider the case of the entry ID when you enter a value in the *Position to* field. The following Validation List Entry commands are accessible from the above list panel:

Opt/F-Key	Command
1	Verify Validation List Entry (VFYVLDLE)
2	Change Validation List Entry (CHGVLDLE)
4	Remove Validation List Entry (RMVLDLE)
5	Display Validation List Entry (DSPVLDLE)
6	Display Validation List Entry (DSPVLDLE OPTION(*PRINT))
F6	Add Validation List Entry (ADDVLDLE)

Entry IDs and *Entry Data* whose length exceeds the space offered in the list panel are indicated by a greater than sign (>) in the rightmost position of the respective field. Positioning the cursor to that field and pressing F22 will display the full field value in the window opening following this action. You can also use function key F1 to take advantage of the online and cursor-sensitive help text panel group for further explanation of the list panel and its various elements.

If you prompt any of the validation list entry commands, you'll notice that I'm storing and using the hexadecimal version of the validation list entry as input to the command in question, as displayed in the example below:

```

                                Display Validation List Entry (DSPVLDLE)

Type choices, press Enter.

Validation list . . . . . > QSASRVID2B      Name
Library . . . . . > QUSRSYS      Name, *LIBL,
*CURLIB
Entry ID:

Entry ID . . . . . > *HEX      Character value,
*HEX
Coded character set identifier *DFT      1-65534, *DFT, *HEX

Output format . . . . . *CHAR      *CHAR, *HEX

Output . . . . . *      *, *PRINT

Entry ID hexadecimal . . . . . > F640404040404040

```

The reason I'm using this approach is due to the behavior employed by the UIM command processor, which replaces values below x'40' with the special value x'1F'. This conversion occurs irrespective of the command being prompted or not, and of course invalidates an entry ID containing any value in the range between x'00' and x'39' excluding x'1F' itself. Another option would be using an exit program to perform the UIM list action and then have the exit program use the Process Command (QCPCMD) API to prompt and process the appropriate validation list entry command. In contrast, the QCPCMD command preserves the accurate value of the entry ID, including values below x'40'.

As mentioned earlier, I've updated the DSPVLDLE command to also show a validation list entry's *Encryption attribute*. It will turn up on the third and last page of the DSPVLDLE command's display panel, as demonstrated in the example below:

```

                                Display Validation List Entry

WYNDHAMW
22-08-09
11:35:38
Validation list . . : QSASRVID2B

```

```
Library . . . . . : QUSRSYS

Encryption attribute . . . . . : 1

Bottom
Press enter to continue.

F3=Exit    F5=Refresh    F12=Cancel    F22=Display entire field value
```

To help you (and myself) keep track of all the validation list commands, and at the same time provide easy access to them, I created the UIM menu CMDVLDL2 which groups and lists all the commands relating to validation lists and validation list entries. Running the command GO CMDVLDL2 from any command line should return the following menu:

```
CMDVLDL2                                Validation List Commands

System:

WYNDHAMW
Select one of the following:

Validation list commands

1. Create Validation List
CRTVLDL
2. Delete Validation List
```

```

DLTVLDL
    3. Clear Validation List
CLRVLDL
    4. Work with Validation List
WRKVLDL
    5. Convert Validation List
CVTVLDL

Validation list entry commands

    11. Add Validation List Entry
ADDVLDLE
    12. Display Validation List Entry
DSPVLDLE
    13. Verify Validation List Entry
VFYVLDLE
    14. Change Validation List Entry
CHGVLDLE
    15. Remove Validation List Entry
RMVLDLE
    16. Work with Validation List Entries
WRKVLDLE

More...
Selection or command

===>

F3=Exit    F4=Prompt    F9=Retrieve    F12=Cancel

```

The menu also offers direct access to all the listed commands' general help text. Simply point your cursor to the menu option of interest and press F1 to learn more about the command in question. I hope you've found the discussion of validation lists and the addition of all the new validation list commands useful. If you have any questions or ideas with respect to this, please don't hesitate to contact me at flensburg@novasol.dk.

This APIs by Example includes the following sources:

```

CBX204    -- RPGLE    -- Display Validation List Entry - CPP
CBX204E   -- RPGLE    -- Display Validation List Entry - UIM General Exit
CBX204H   -- PNLGRP   -- Display Validation List Entry - Help
CBX204P   -- PNLGRP   -- Display Validation List Entry - Panel Group
CBX204V   -- RPGLE    -- Display Validation List Entry - VCP
CBX204X   -- CMD      -- Display Validation List Entry

CBX2061P  -- PNLGRP   -- Work with Validation Lists - Panel Group

CBX207    -- RPGLE    -- Work with Validation List Entries - CCP

```

```
CBX207E  -- RPGLE  -- Work with Validation List Entries - UIM Gen.  
Exit  
CBX207H  -- PNLGRP -- Work with Validation List Entries - Help  
CBX207L  -- RPGLE  -- Work with Validation List Entries - UIM List  
Exit  
CBX207P  -- PNLGRP -- Work with Validation List Entries - Panel Group  
  
CBX207V  -- RPGLE  -- Work with Validation List Entries - VCP  
  
CBX207X  -- CMD     -- Work with Validation List Entries  
  
CBX207U  -- UIM     -- Validation List Commands Menu  
  
CBX207M  -- CLP     -- Validation List Commands - Build commands
```

Note that I've corrected an error in the Work with Validation Lists list panel group published last time and consequently I've had to include a revised version with this article's code download . The old version would not be able to call the WRKVLDE command published today correctly, so please be sure to download the new version to the appropriate source file before you attempt to run today's CBX207M CL command build program, which will ensure that the CBX207P panel group gets re-created and works correctly.

To create all the above Validation List command objects, compile and run CBX207M, following the instructions in the source header. Note that for the CL program CBX207M to run successfully, and for the new validation list command objects to compile, you'll need the CBX205M CL program included with the July 25, 2009, APIs by Example article. As always, the compilation instructions are also included in the respective source headers.

IBM Documentation:

[Validation List Objects](#)

[Planning the Use of Validation List Objects](#)

[Validation list on HTTP Server](#)

Previously published related articles:

[APIs by Example: Adding to the Collection of Validation List Commands](#)

[APIs by Example: A Validation List Entry's Life Cycle in CL Commands](#)

[APIs by Example: Have a Peek at Validation List Entries](#)

[APIs by Example: User Function Registration APIs, Part 1](#)

[APIs by Example: User Function Registration APIs, Part 2](#)

[APIs by Example: User Function Registration APIs, Part 3](#)

[APIs by Example: Validation List APIs](#)

[APIs By Example: Profile Authorization Management](#)

[APIs by Example: Cryptographic Services APIs, Part 3](#)

[APIs by Example: Cryptographic Services APIs, Part 7](#)

This article demonstrates the following Validation List APIs:

[Find Validation List Entry API - QsyFindValidationLstEntry\(\)](#)

[Open List of Validation List Entries \(QSYOLVLE\) API](#)

[Validation List APIs](#)

[Digital Certificate Management APIs](#)

[Retrieve the source code for this API example.](#)

Source URL: <http://iprodeveloper.com/rpg-programming/apis-example-listing-and-working-validation-list-entries>