CL and Command Programming Lab Exercise

By Bob Bittner

IBM eServer Solutions Enablement

June 2004

Overview

This lab exercise will introduce you to CL programming on OS/400®. Be sure to first view the prerequisite Web presentations as listed before doing this exercise.
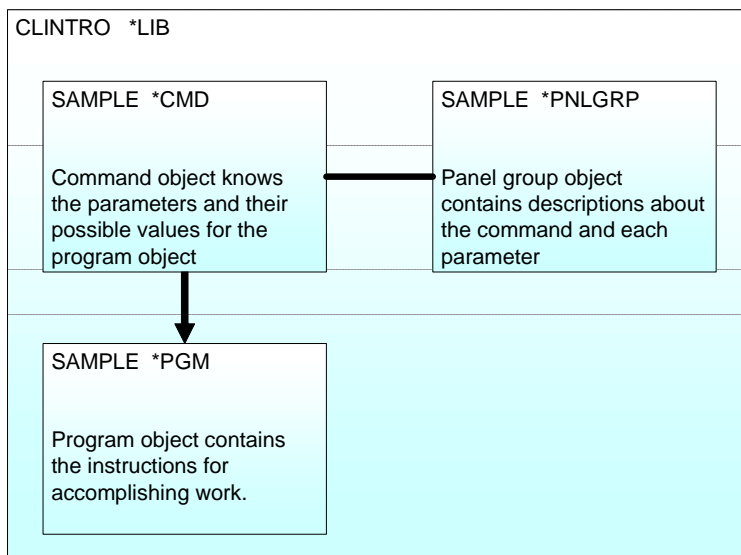
Prerequisites

5250 Terminal Emulation Text Interface online course
(**ibm.com**/servers/enable/site/education/abstracts/term5250_abs.html)
Introduction to CL and Command Programming online course (URL to come soon)

You will create three objects during this lab exercise. First, you will create and test a simple CL **program** object. Next, to more easily call the program, you will create a **command** object. Finally, you will create a **panel group** object for the help text of the command.

In review, here are the objects and how they interact:

CLINTRO *LIB

SAMPLE *CMD	SAMPLE *PNLGRP
Command object knows the parameters and their possible values for the program object	Panel group object contains descriptions about the command and each parameter

SAMPLE *PGM

Program object contains the instructions for accomplishing work.

1. At the command line, a user types the name of a command object and presses F4.
2. The command object references the panel group object for displaying help text.
3. Once the parameters are entered, the system calls the program object specified in the command object.

Overview of lab exercise steps

Here is an overview of the exercises that we will cover in this lab.

Entry and Setup
- Sign on as your specific User Profile
- Create a library named CLINTRO
- Change current library to CLINTRO

CL Program
- Create a source physical file for the CL program
- Enter (or copy / paste) the CL program source code
- Compile CL source code into a program
- Call the program

CL Command
- Create a source physical file for the CL command
- Upload the CL command source code using FTP
- Compile command source code into a CL command
- Run the command

Panel Group (Help Text)
- Create a source physical file for the panel group containing help text
- Upload the panel group source code using FTP
- Compile panel group source code into a panel group object
- Prompt on the command, see help text

Entry and Setup

You will need a user profile on your IBM® eServer® iSeries™ server. Your user profile does not need any OS/400 special authorities. The tasks you will need to perform are available to all users on a server as shipped from the factory.

From the sign-on display, enter your user name and password.

Create a library named CLINTRO. The following command will create the library CLINTRO.

CRTLIB CLINTRO

Now, change your current library to CLINTRO. This step is to provide a default location for creating new objects and for the compilers to find source code. The following command will change your current library to CLINTRO

CHGCURLIB CLINTRO

Creating the CL Program

CL source code is stored as a *member* of a *source physical file.* Multiple source members can reside in a single source physical file. To create a source physical file, enter the following command. By default, the file will be created in the *current library*.

CRTSRCPF QCLSRC

Once you have created the source physical file, use Programming Development Manager (PDM) to see the members within the file. PDM is a collection of tools to work with objects, source code, etc. Enter the following command to start PDM at the "source members" level.

WRKMBRPDM CLINTRO/QCLSRC

This will display a list of source members in the file QCLSRC in the library CLINTRO. Since you just created the source file, there are no members in it.

```
                    Work with Members Using PDM                 S105NB9M

 File  . . . . . .    QCLSRC
   Library . . . .     CLINTRO            Position to  . . . . . _____

 Type options, press Enter.
  2=Edit        3=Copy  4=Delete 5=Display      6=Print     7=Rename
  8=Display description  9=Save  13=Change text  14=Compile  15=Create module...

 Opt  Member      Type        Text

     (No members in file)



                                                                 Bottom
 Parameters or command
 ===>
 F3=Exit         F4=Prompt          F5=Refresh          F6=Create
 F9=Retrieve     F10=Command entry  F23=More options    F24=More keys
                                     (C) COPYRIGHT IBM CORP. 1981, 2000.


```

NOTE: in the list of function key descriptions, F6=Create.
Press **F6** to prompt for parameters to creating a new source member.

```
                    Start Source Entry Utility (STRSEU)

 Type choices, press Enter.

 Source file  . . . . . . . . . > QCLSRC        Name, *PRV
   Library  . . . . . . . . . . >   CLINTRO     Name, *LIBL, *CURLIB, *PRV
 Source member  . . . . . . . .   SAMPLE        Name, *PRV, *SELECT
 Source type  . . . . . . . . .   CLP           Name, *SAME, BAS, BASP...
 Text 'description' . . . . . .   Sample CL program




                                                                    Bottom
  F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
  F24=More keys
```

Fill in the parameters as shown above. Specifying "CLP" for the source type informs the editor that this is to be CL program source code. As such, it will allow command prompting just as if entering a command on the command line. The editor has the incredibly creative name, Source Entry Utility (SEU). When you press **Enter**, a new member will be created.

Now, you are editing the source physical file member, CLINTRO/QCLSRC.SAMPLE. Locate the CL program source code below and copy the first 15 lines to the clipboard. Since the paste function only can paste one screen at a time, you need to paste in the source code in two steps.

```
                       Work with Members Using PDM
 Columns . . . :    1  71           Edit                    CLINTRO/QCLSRC
 SEU==>                                                               SAMPLE
 FMT **  ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
        *************** Beginning of data **********************************
'''''''            PGM       PARM(&FILE)
'''''''            DCL       VAR(&FILE) TYPE(*CHAR) LEN(10)
'''''''            DCL       VAR(&USER) TYPE(*CHAR) LEN(10)
'''''''            DCL       VAR(&REPLY) TYPE(*CHAR) LEN(1)
'''''''
'''''''            RTVJOBA   CURUSER(&USER)
''''''' TOP:
'''''''            SNDPGMMSG  MSG(&USER *TCAT ' RUNNING REPORT WITH OUTPUT +
'''''''                         FILE SET TO ' *CAT &FILE)
'''''''        /* CALL      PGM(REPORT) PARM(&FILE)      */
'''''''            MONMSG    MSGID(CPF0000) EXEC(GOTO ERR)
'''''''            SNDPGMMSG  MSG('COMPLETED RUNNING REPORT TO ' *CAT +
'''''''                         &FILE) MSGTYPE(*COMP)
'''''''            GOTO END
''''''' ERR:      SNDUSRMSG  MSG('FAILURE RUNNING REPORT TO ' *CAT &FILE +
        ***************** End of data **********************************
```

After pasting in the first 14 lines, copy the remaining source. After pressing **Page Down**, there are not blank lines into which to paste the code. You need to tell the editor to add some lines. Type **I10** over the numbers on the left and press enter. Then, paste the remaining source code into the editor.

```
                         Work with Members Using PDM
Columns . . . :    1  71              Edit                       CLINTRO/QCLSRC
 SEU==>                                                                     SAM2
 FMT **   ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0009.00                             FILE SET TO ' *CAT &FILE)
0010.00          /*  CALL       PGM(REPORT) PARM(&FILE)      */
0011.00              MONMSG      MSGID(CPF0000) EXEC(GOTO ERR)
0012.00              SNDPGMMSG   MSG('COMPLETED RUNNING REPORT TO ' *CAT +
0013.00                          &FILE) MSGTYPE(*COMP)
I10 .00              GOTO END
       ***************** End of data ****************************************



 F3=Exit   F4=Prompt    F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
 F16=Repeat find        F17=Repeat change          F24=More keys
```

Now you are finished entering the source code for the CL program. Press **F3** to exit the editor. You will be prompted for some options. Just press **Enter**. If you are returned to edit the source code, there was an error in copying the source, and the editor is telling you that there are syntax errors.

That's enough usage of SEU. Just know that it is there to edit source physical file members. It is handy as an editor for these programs, since you can press F4 right on the command you are editing and prompt just like on the command line. But, its other features leave plenty to be desired. That's why IBM created more user-friendly, graphical tooling in WebSphere Development Studio Client. This is a client/server program that runs on PC clients and works more like a normal editor in that environment.

You should be back at the "Work with Members Using PDM" screen.

Compile the CL source to create a program object. *You can use the "14" option of this display to compile/create a CL program. By default, this will submit the compile to batch, rather than run the compile in your current job.* Instead, use the following CL command to compile the CL source into a program in one step.

Type the command below and press **F4**. Examine the parameter values and press **Enter**. This works because your source file name is QCLSRC, the system-expected default name. This will create the program SAMPLE in the current library (CLINTRO).

CRTCLPGM SAMPLE

You should see a message "Program SAMPLE created in library CLINTRO".

DEBUG:
*If you receive a message that the program did not compile, you need to look at a spooled file for your job. Simply enter the command **WRKSPLF** to find a list of output for your user profile. Locate the file named SAMPLE and use option **5** to display the listing.*

Call the SAMPLE program with parameter "ABC"
Since the *current library* is part of the *library list*, to run the program, simply call it by name. The system will find the program SAMPLE, recently created in your CLINTRO library. Enter the following command.

CALL SAMPLE PARM(ABC)

Creating the CL Command

You just created a CL program object. From the command line, you can only enter the name of a command object. One of these commands is CALL. As you just did, you can use the CALL command to run a program. CALL does not know what parameters a program needs.

To be able to simply enter SAMPLE on the command line, you need to create a CL command object named SAMPLE. When creating the SAMPLE command object, you will define the program's parameters.

CL command source code is stored as a *member* of a *source physical file*, just like the CL program source code before this and the help text panel group source code after this. Again, first create a source physical file object called QCMDSRC. By default, it will be created in the current library, which should still be CLINTRO.

CRTSRCPF QCMDSRC

Last time, you used PDM and SEU to get the source code onto the server. To learn another way to get the source code onto your server, use FTP from your client. The CL command source code is pretty short. Copy the CL command source code to your text editor of choice and save the file locally as **sample.cmd**. Then, follow the series of FTP commands shown below. You might see an error message from the first command. If so, ignore it.

```
C:\TEMP>ftp yoursystem
Connected to yoursystem.yourcompany.com.
220-QTCP at yoursystem.yourcompany.com.
220 Connection will close if idle more than 5 minutes.
User (yoursystem.yourcompany.com.:(none)): yourprofile
331 Enter password.
Password:
230 YOURPROFILE logged on.
ftp> cd /qsys.lib/clintro.lib
ftp> quote site namefmt 0
ftp> cd clintro
ftp> put sample.cmd qcmdsrc.sample
ftp> quit
```

The OS/400 FTP server can use two different name formats. The "put" command above shows using name format 0, which is *fileobject.member*. If your FTP server's attribute "Initial name format" is set to *LIB, you can skip the first two of these FTP commands.

Now that you have the CL command source where it needs to be, you can create the CL command object. The command below will accomplish this. Again, the CL command object will be created in the current library, which should still be CLINTRO.
Type the command below and press F4. Examine the parameter values and press Enter.

CRTCMD CMD(SAMPLE) PGM(SAMPLE)

```
                          Create Command (CRTCMD)

 Type choices, press Enter.

 Command  . . . . . . . . . . . . > SAMPLE        Name
   Library  . . . . . . . . . . .     *CURLIB     Name, *CURLIB
 Program to process command . . . > SAMPLE        Name, *REXX
   Library  . . . . . . . . . . .     *LIBL       Name, *LIBL, *CURLIB
 Source file  . . . . . . . . . .   QCMDSRC       Name
   Library  . . . . . . . . . . .     *LIBL       Name, *LIBL, *CURLIB
 Source member  . . . . . . . . .   *CMD          Name, *CMD
 Threadsafe . . . . . . . . . . .   *NO           *YES, *NO, *COND




                                                                    Bottom
 F3=Exit    F4=Prompt    F5=Refresh    F10=Additional parameters   F12=Cancel
 F13=How to use this display        F24=More keys


```

Specifying only the name parameter works because your source file name is QCMDSRC.
QCMDSRC is the system-expected default name for this command.

When you press **Enter**, you should see a message "Command SAMPLE created in library
CLINTRO".

DEBUG:
*If you receive a message that the program did not compile, you need to look at a spooled file for
your job. Simply enter the command **WRKSPLF** to find a list of output for your user profile.
Locate the file named SAMPLE and use option 5 to display the listing.*

Run the SAMPLE command with parameter "FILE(ABC)"
The *current library* is part of the *library list*. The command interpreter looks for command objects
in your library list. To run the command, simply enter its name.

Enter the following command.

SAMPLE FILE(ABC)

Creating the Panel Group (Help Text)

Once you have a command object, it would be handy to enable help text for the command and each parameter. The help text is contained in a panel group object.

Just like for the previous two objects, the source code for the panel group object is stored as a *member* of a *source physical file*. By default, the file will be created in the *current library*. Enter the following command to create a source physical file.

CRTSRCPF QPNLSRC

Fire up your editor of choice (locally) and copy the source code below for the panel group into your editor. Modify some of the text, leaving the obvious tags intact. Be creative. Save the file as **sample.pnl**. Again, use FTP to put the source code on the server.

```
C:\TEMP>ftp yoursystem
Connected to yoursystem.yourcompany.com.
220-QTCP at yoursystem.yourcompany.com.
220 Connection will close if idle more than 5 minutes.
User (yoursystem.yourcompany.com.:(none)): yourprofile
331 Enter password.
Password:
230 YOURPROFILE logged on.
ftp> cd /qsys.lib/clintro.lib
ftp> quote site namefmt 0
ftp> cd clintro
ftp> put sample.pnl qpnlsrc.sample
ftp> quit
```

Take a look at the source code on the server using PDM and SEU, as shown below. Enter the following command.

WRKMBRPDM QPNLSRC

```
                         Work with Members Using PDM               S105NB9M

 File  . . . . . .    QPNLSRC
   Library . . . .      CLINTRO            Position to  . . . . .  _____

 Type options, press Enter.
  2=Edit         3=Copy  4=Delete  5=Display       6=Print     7=Rename
  8=Display description  9=Save  13=Change text  14=Compile  15=Create module...

 Opt  Member      Type        Text
  5   SAMPLE      _____    _____



                                                                    Bottom
 Parameters or command
 ===>
 F3=Exit         F4=Prompt           F5=Refresh          F6=Create
 F9=Retrieve     F10=Command entry   F23=More options    F24=More keys

```

Now you are looking at members of your QPNLSRC file.
Display the file member using option 5 on this screen.
Type **5** on the line next to the file member SAMPLE and press **Enter**.
Press **F12** to cancel the display.

Compile the panel source to create a panel group object. *You can use the "14" option of this display to compile/create the panel group. By default, this will submit the compile to batch, rather than run the compile in your current job.* Instead, use the following CL command to compile the panel group source into a panel group object. This will create the command SAMPLE in the current library. Type the command below and press **F4**.

CRTPNLGRP SAMPLE

```
                       Create Panel Group (CRTPNLGRP)

 Type choices, press Enter.

 Panel group  . . . . . . . . . . >  SAMPLE        Name
   Library  . . . . . . . . . . .      *CURLIB     Name, *CURLIB
 Source file  . . . . . . . . . .     QPNLSRC      Name
   Library  . . . . . . . . . . .      *LIBL       Name, *LIBL, *CURLIB
 Source member  . . . . . . . . .     *PNLGRP      Name, *PNLGRP
 Text 'description' . . . . . . .     *SRCMBRTXT   _____




                                                                    Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display       F24=More keys

```

Examine the parameter values and press **Enter**.

Note, again, that there is a system-expected default name for the source file. This is QPLSRC, the same as you created and are using.

You should see a message "Panel group SAMPLE created in library CLINTRO".

DEBUG:
*If you receive a message that the program did not compile, you need to look at a spooled file for your job. Simply enter the command **WRKSPLF** to find a list of output for your user profile. Locate the file named SAMPLE and use option **5** to display the listing.*

Change the SAMPLE command to point at the newly created panel group for help text. Enter the following command:

CHGCMD SAMPLE HLPPNLGRP(*LIBL/SAMPLE) HLPID(SAMPLE)

Note that the help panel group is specified as *LIBL/SAMPLE. As such, the help text will only be found when the SAMPLE panel group is in a library in the library list.

You are done building objects! Now it's time for the fun part, to see the fruits of this exercise.

Prompt on the SAMPLE command.

Type **SAMPLE** and press **F4**.

Press **F1** with the cursor on various parts of the screen.

Congratulations, NOW you are a CL programming guru!

CL Program source code

```
           PGM         PARM(&FILE)
           DCL         VAR(&FILE) TYPE(*CHAR) LEN(10)
           DCL         VAR(&USER) TYPE(*CHAR) LEN(10)
           DCL         VAR(&REPLY) TYPE(*CHAR) LEN(1)

           RTVJOBA     CURUSER(&USER)
TOP:
           SNDPGMMSG   MSG(&USER *TCAT ' running REPORT with output +
                         file set to ' *CAT &FILE)
        /*  CALL        PGM(REPORT) PARM(&FILE)       */
           MONMSG      MSGID(CPF0000) EXEC(GOTO ERR)
           SNDPGMMSG   MSG('Completed running REPORT to ' *CAT +
                         &FILE) MSGTYPE(*COMP)
           GOTO END
 ERR:      SNDUSRMSG   MSG('Failure running REPORT to ' *CAT &FILE +
                         *TCAT '. Would you like to try it again? +
                         (Y or N)') VALUES(Y N) DFT(N) +
                         MSGTYPE(*INQ) MSGRPY(&REPLY)
           IF          COND(&REPLY *EQ 'Y') THEN(GOTO TOP)
           SNDPGMMSG   MSGID(CPF0006) MSGF(QCPFMSG)
END:       ENDPGM
```

CL Command source code

```
           CMD         PROMPT('Sample Command')
           PARM        KWD(FILE) TYPE(*CHAR) LEN(10) DFT(MYOUTPUT) +
                         PROMPT('Output file name')
```

Panel Group source code

```
:PNLGRP.
:HELP NAME=SAMPLE.
:P.
This text describes the SAMPLE command.  Normally, this
would include an overall description of what the command
does.  Do not include descriptions of the individual
parameters here.  Displaying Extended Help will do so.
Here's an example of a :HP2.highlighted phrase.:EHP2.
Other formatting tags are explained in the book
:HP1.Application Display Programming,:EHP1. SC41-5715.
:EHELP.
:HELP NAME='SAMPLE/FILE'.
:H3.Output File (FILE)
:P.
Text entered here describes the FILE parameter.
This is the place to describe any reserved values for the parameter.
You can span multiple lines, of course.  Formatting is adjusted by
the system.
See
what
I
mean?
:EHELP.
:EPNLGRP.
```

Trademarks