Lab Objectives

- **Look at commands that are similar to the ones required to complete lab 4**
- **Select the appropriate Lab 4 command and parameter after looking at similar commands**

Part A – Investigating Lab4 commands

RTVJOBA

Open your CLLE partially done program and type the following:

When (&Option = '3' *or &option = '03') Do

RtvJobA   Press the prompt key (F4) and examine the possible information you can retrieve from a running job. (we are going to try getting info about the output queue)

Retrieve Job Attributes (RTVJOBA)		
CL var for INQMSGRPY (10):		Character value
CL var for STSMSG (7):		Character value
CL var for BRKMSG (7):		Character value
CL var for DEVRCYACN (13):		Character value
CL var for RTNCDE (5 0):		Number
CL var for ENDSTS (1):		Character value
CL var for PRTDEV (10):		Character value
CL var for OUTQ (10):	> &OUTQ	Character value
CL var for OUTQLIB (10):	&OUTQLIB	Character value
CL var for SPLFACN (10):		Character value
CL var for PRTTXT (30):		Character value
CL var for PRTKEYFMT (10):		Character value
CL var for SRTSEQ (10):		Character value
CL var for SRTSEQLIB (10):		Character value
CL var for LANGID (3):		Character value
CL var for CNTRYID (2):		Character value
CL var for CCSID (5 0):		Number
CL var for DFTCCSID (5 0):		Number
CL var for CHRIDCTL (10):		Character value
CL var for DECFMT (1):		Character value
CL var for DATFMT (4):		Character value
CL var for DATSEP (1):		Character value
CL var for TIMSEP (1):		Character value
CL var for DATE (6):		Character value
CL var for CYMDDATE (7):		Character value
CL var for DATETIME (20):		Character value
CL var for DAYOFWEEK (4):		Character value
CL var for TIMZON (10):		Character value

☐ Advanced   ☐ All Parameters ☐ Keywords

RTVJOBA OUTQ(&OUTQ) OUTQLIB(&OUTQLIB)

| OK | Restore defaults | Cancel |

The keyword notation of how to use this command appears at the bottom of this screenshot. That is the code that will be placed in your program.

In Lab 4a, we had a subroutine executed when someone selected option 1. This time we will show sample code (with a different command featured then the actual lab) that will determine the output queue for the current job. This sample code will provide a solution inline, without using a subroutine.

 Let's pretend that this is option three.

```
        Select
        When (&Option = '1' *or &Option = '01')        CallSubr OPTION1
        ..
        When(&Option = '3' *or &Option = '03')         DO

        RTVJOBA    OUTQ(&OUTQ) OUTQLIB(&OUTQLIB)
        /* note this shows at the bottom of the preceding screen shot */

        CHGVAR    &MsgTxt     ('Output queue for the current job is'     +
           *Bcat &OUTQ
           *Bcat 'in the '
           *Bcat   &OUTQLIB *Bcat 'Library')

         EndDo
         When( …)
        …
        Otherwise        ChgVar &in30 '1'
        EndSelect
```

This Select statement needs some added support at the top of your program. There are two new fields that need to be declared. The screenshot for the RTVJOBA command lets you know the type and size required for these declarations. What are the required declarations?

_____

_____

The result of selecting option three would show something like the following:

Output queue for the current job is WS433D40 in the WS433D40 Library

RTVUSRPRF

You can get information about the user profile for the current user that is signed in to an interactive session.
The Retrieve User Profile command can tell information like the "initial menu" setting for a user.

We are going to prompt the command to find out all the useful user profile information that is retrievable.

Retrieve User Profile (RTVUSRPRF)

CL var for PWDEXPITV (5 0):		Number
CL var for PWDEXP (4):		Character value
CL var for STATUS (10):		Character value
CL var for PRVSIGN (13):		Character value
CL var for NOTVLDSIGN (11 0):	&NOTVLDSIGN	Number
CL var for LMTDEVSSN (7):		Character value
CL var for KBDBUF (10):		Character value
CL var for LANGID (10):		Character value
CL var for CNTRYID (10):		Character value
CL var for CCSID (5 0):		Number
CL var for SRTSEQ (10):		Character value
CL var for SRTSEQLIB (10):		Character value
CL var for OBJAUD (10):		Character value
CL var for AUDLVL (640):		Character value
CL var for GRPAUTTYP (10):		Character value
CL var for SUPGRPPRF (150):		Character value
CL var for UID (10 0):		Number
CL var for GID (10 0):		Number
CL var for SETJOBATR (160):		Character value

☐ Advanced   ☐ All Parameters ☐ Keywords

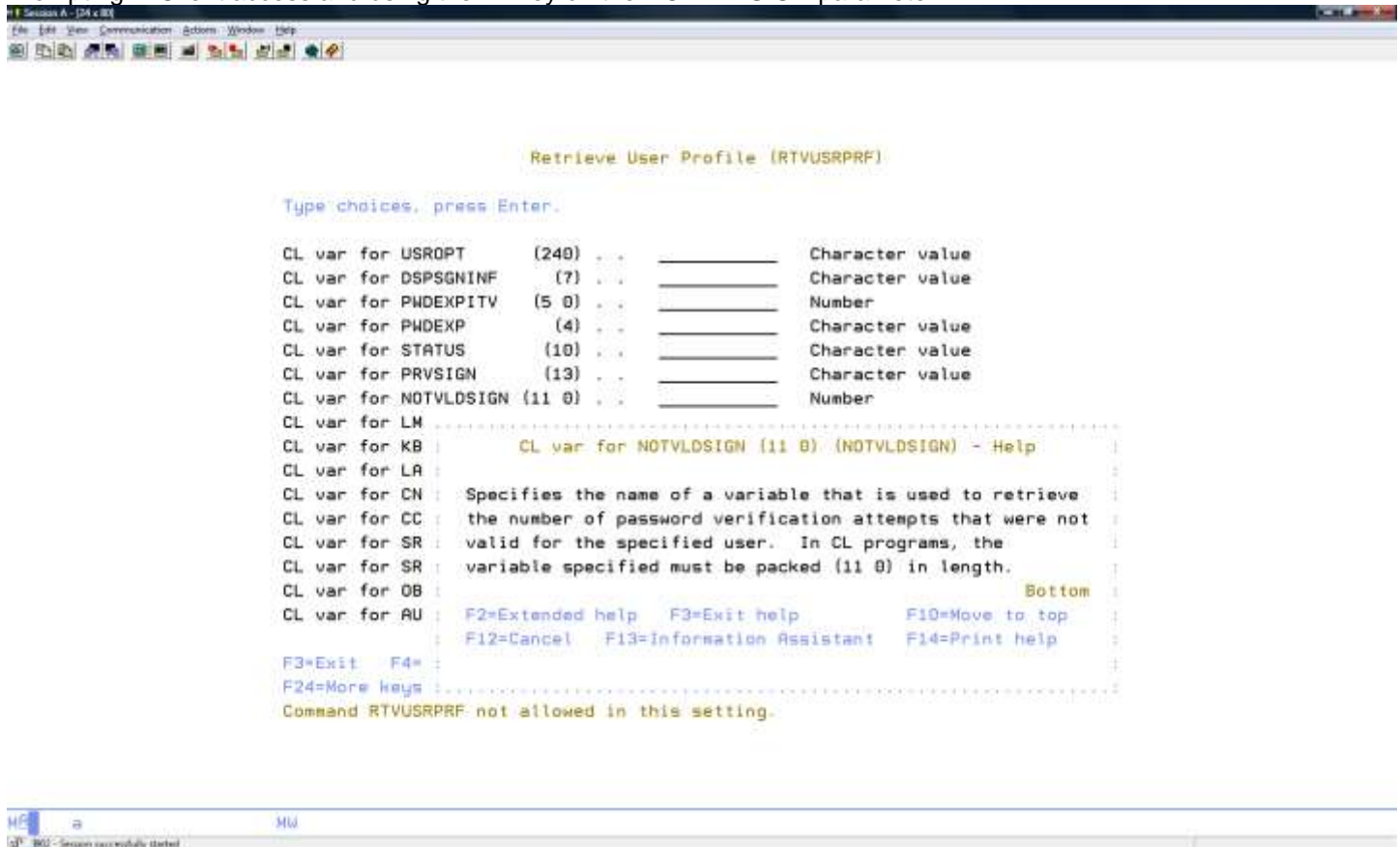RTVUSRPRF NOTVLDSIGN(&NOTVLDSIGN)

[OK] [Restore defaults] [Cancel]

Note the actual command shows at the bottom of the screenshot again.

This time we are not sure what NOTVLDSIGN is telling us. The F1 key does not supply immediate context sensitive help.

You can try prompting the command in Client Access and with your cursor on the appropriate parameter, get more information about that parameter.

Prompting in Client access and using the F1 key on the NOTVLDSIGN parameter.

Here is the command used in a select statement:

```
When (&Option = '4' *or &option = '04') Do
        RTVUSRPRF  NOTVLDSIGN(&NOTVLDSIGN)
        ChgVar &NotVldSC &NotVldSign
        ChgVar    &MsgTxt ('Not Valid Signon number is ' *Cat &NotVldSC)
ENDDO
```

Show the two fields need to be declared at the top of the program to support option 4.

_____

_____

RTVSYSVAL.

You can use the same strategies explored with RTVJOBA and RTVUSRPRF to investigate system values

This program requires retrieving a system value, user profile information, job attributes for options 1 to 5. Option 6 involves sending a message and using the message queue and message text fields that appear at the bottom of the screen.

Making the Command Line available while your program is running.

Your screen record needs allow the user to press the F21 key.

Your program can check for this key being used somewhere in your Select statement. The API program that will support the use of a command line while your program is running is QUsCmdLn. You just need to call this at the appropriate time.

Looping

A good model to follow with reading files and presenting interactive screens is the following:

SNDRCVF

DoWhile /* (something to do with the F3 key) */

….

        SNDRCVF
ENDDO
/* summary or cleanup operations */

Initializing

You probably need to reset fields at some point in your loop. It is suggested that most of this can be done in an Initialize subroutine.