

IBC233 - System i Business Computing

Week 2: Library List, RDp and PF

Agenda

- ▶ Week 1 review
- ▶ System Values
- ▶ Library List
- ▶ Intro to RDp
- ▶ Physical File
- ▶ Lab2

Week 1/Lab 1 Review

- ▶ Function keys used in Lab1
 - F1, F3, F4, F9 , F6, F12
- ▶ Commands used in Lab1
 - DSPLIBL, CALL LIBRARY/PROGRAM, QSHELL, GO MAIN, GO PROGRAM, DSPMSG, WRKSPLF, WRKOBJPDM, SIGNOFF, ...
- ▶ Input a number field
 - Positive number: type numbers + plus sign
 - Negative number: type numbers + minus sign
- ▶ Exit from a never ending program
 - Shift+ESC (then 2, or enter or 7)

Week 1/Lab 1 Review

- ▶ Command to Display Library List
 - DSPLIBL
- ▶ Library types in a library list
 - System library
 - User library
 - Current library
 - Product library
- ▶ Command to Display Library List for a specific library type – using system values
 - DSPSYSVAL QSYSLIBL
 - DSPSYSVAL QUSRLIBL
- ▶ Show other system value
 - DSPSYSVAL QMODEL

Week 1/Lab 1 Review

► About source physical file

- Source code is stored in: a **source physical file** (a kind of container).
 - To create a source physical file, use command:
 - **CRTSRCFP FILENAME**
 - e.g. CRTSRCFP QCLLESRC
 - A source physical file can contain many source programs as members.
 - To create a source program within a source physical file
 - use command **WRKMRBPDM FILENAME**, then F6
 - name source program member: e.g. SYSVALPRG
 - To create Another source program
 - e.g. MYPRG2

Week 1/Lab 1 Review

- ▶ About CL
 - What does CL stand for?
 - CL program type: CLLE
 - CL Code is between : PGM and ENDPGM
- ▶ To work with compiled CL program - *PGM object:
 - WRKOBJPDM
- ▶ To see your compiled listing
 - WRKSPLF
- ▶ What is QGPL? Give an example when do you need it.
 - e.g. moving the spooled file from your personal output queue to an output queue used for printing out spooled files.

Definition Review

Power Systems	Unix/Linux Equivalent or Definition
User Id / User Profile	Account
Library	Similar to a directory
Interactive Job	What you do from the time that you sign on to when you sign off
Batch Job	Work that you submitted to the background
Queue	Line up
Objects	Everything that takes up space, has a description and is not temporary
Files	Store Data ONLY !!! (different from Unix)
Members	Partition of a file

File vs Member

- ▶ In System i, file and member are confusing concepts. we cannot find equivalent definitions in Unix or Windows.
- ▶ Source file, or **source physical file** can be viewed as a container which stores members.
 - e.g. you may create a source physical file, called Q**CLLESRC**, to store all your **CL** programs.
 - e.g. you may create a source physical file, called Q**RPGLESRC**, to store all your **RPG** programs.
- ▶ Member is a coded program, or a group of records which is stored in a source physical file.
 - e.g. in Lab 1, source code SYSVALPRF is a member of **file** or **source physical file** QCLLESRC.
 - e.g. in Lab 2, **physical file** STUDENTS (even though it sounds like a file) is a member of the **source physical file** QDDSSRC.

System Values vs User Profile

- ▶ System Values configures ZEUS for everyone.
- ▶ User Profile customizes ZEUS's configuration for an individual

What's ZEUS Doing?

- ▶ WoRK with ACTive JOBs
- ▶ System vs Subsystem
 - Subsystem
 - ▶ Independent predefined operation environment.
 - ▶ Coordinates the work flow and resource use.
 - ▶ All jobs, with the exception of system jobs, run within subsystems.
 - ▶ e.g. QINTER and QBATCH
- ▶ Effects of an infinite loop

How the System i finds Objects

- ▶ When an object is created, it is created 'in' a particular library.
- ▶ To access an object, the iSeries needs to know what library the object is in.
- ▶ (CL) Commands prompt you for the object name *and library name*.
- ▶ ***LIBL** is the default for library

Library Lists

- ▶ The list of Libraries through which the iSeries searches to find objects.
- ▶ Similar to the DOS 'Path' concept
- ▶ Each Job has a library list
- ▶ ***LIBL** = library list

Library Lists

- A Library List consists of 4 parts:
 - System Libraries (up to 15)
 - Product Libraries (none, 1 or 2)
 - Current Library(1 only)
 - User Library (up to 25)
- To view your library list – DSPLIBL
- To add a library to library lists
 - e.g. ADDLIBLE IBC233LIB

Using Library List

- ▶ Example: EDITCODES is a program in Library IBC233LIB. When running the program (CALL EDITCODES), we get the message:
 - “Program EDITCODES in Library *LIBL not found”.
- ▶ Solution:
 1. CALL IBC233LIB/EDITCODES
 2. ADDLIBLE IBC233LIB
CALL EDITCODES
- ▶ Note:
 - Library lists are built when you sign on System i and destroyed when signoff.
 - ADDLIBLE command can only add a library to the list for the current session; the library you added will no longer on the list after your next sign on.

How do Libraries get on your Library List ?

- When you sign on, your library list is built from:
 - QSYSLIBL (system value)
 - QUSRLIBL (system value)
 - Current library from User Profile
- Product libraries are added as needed by the system

Storing Objects

- ▶ When an object is created, it is stored in the library specified.
- ▶ If no library is specified, it is stored in ***CURLIB** (current library).
- ▶ If there is no current library, it is stored in **QGPL** library.
- ▶ Exception: Library Descriptions, Device Descriptions and User Profiles are always stored in library **QSYS**.

Languages that we'll be Learning

► CL (Control Language)

- How we communicate with the operating system
- Similar idea to Unix Schell

► RPG (Report Program Generator)

- How we **access** the database
- C does the same thing

► DDS (Data Description Specifications)

- How we **define** the Database

► CMD – The language we use to write commands

Steps to Writing a CL Program



Writing a CL program

*OUTQ

WRKSPLF

Source Code
QCLLESRC
*FILE
(PGM1)
Type: CLLE

CL
Compiler

Successfully
Completed

Compiled
Listing
(PGM1)

PGM1
*PGM

*MSGQ
DSPMSG

What does RPG Stand for?

Steps to Writing a RPG Program



RDp

RDp - Rational Developer for Power Systems

- ▶ Formerly RDi - Rational Developer for System i
- ▶ Based on Eclipse Technology
- ▶ part of IBM's suite of products designed to enhance programmer's productivity
- ▶ PC version of PDM

Using RDp

- ▶ Tutorial: under “Help” item
 - IBM i RSE Getting Started or Welcome
- ▶ Important operations: under “Window” item
 - Reset Perspective, Show view, ...
- ▶ Workspace
 - After labs, backup workspace and shutdown the PC.
- ▶ Connection: zeus.senecac.on.ca
- ▶ Local (PC) Files vs IFS (System i) files
 - IBC233 course materials: [IFS](#) / [Root file system](#) / [Russ](#) / [IBC233](#)
 - Copy and paste within RDp

Using RDp

- ▶ Using RDp as PDM
 - Expanding ***Work with libraries...***, ... under ***Objects***
- ▶ Library List
 - ***Adding Library List Entry...***
- ▶ Initial Library List & Initial Command
 - ***Objects*** -> property -> ...
- ▶ WRKACTJOB, DSPSPLF in RDp

Using RDp

► Writing a CL program using RDp

- Your current lib -> New -> Source Physical file... ->
- ...
- You have to run the program in Client Access.

► Writing a RPG program using RDp

What does DDS Stand for?

DB2 – the database

- ▶ The database – DB2 is built into IBM i.
- ▶ Database objects are created using either DDS coded programs or SQL.
- ▶ DB2 consists of
 - Physical file (also called tables)
 - Logical files (also called indexes or views)

Data Description Specifications

- ▶ DDS is used to define data.
- ▶ DDS is a sensitive language. Everything must be typed in uppercase.
- ▶ DDS program structure:
 - start out with file level keywords, or attributes that apply to a file itself.
 - **Record** format means a layout or a screen.
 - The fields are then listed
 - At end, the access path (**key**) information.

DDS program example

- This DDS code will create a physical file to store Account Information.

```
T.Name+++++RLen++TDp.....Functions+++++++
UNIQUE
TEXT ('ACCOUNT INFORMATION')_
COLHDG ('ACCOUNT' 'NUMBER')_
COLHDG ('ACCOUNT' 'NAME')_
COLHDG ('ACCOUNT' 'DATE')_
COLHDG ('AMOUNT' 'OWING')_
K ACCT
```

R **ACCTPFR** _____

— ACCT _____ 3**S** 0

— ACCTNAME _____ 30**A**

— ACCTDATE _____ **L**

— AMTOUT _____ 7**P** 2

- **ACCTPFR** is the **record format name**
- **ACCT** is the **key field**

Field Data Types

Entry	Meaning	
P	Packed Decimal	Takes up less space in memory than a Zoned Decimal (A Packed Decimal stores two digits in one byte)
S	Zoned Decimal	If you leave the number of decimal places blank, then none are assumed.
B	Binary	
F	Floating-Point	
A	Character	If you leave the data type blank, then character is assumed.
H	Hexadecimal	
L	Date	
T	Time	
Z	TimeStamp	

* We don't often see Binary, Floating-Point and Hexadecimal fields in business applications

Writing a DDS Program Using RDp

- ▶ Create a Source Physical File (may named Q**DDS**SRC) as a container for the DDS code.
- ▶ Create a new member called ACCOUNTS and its member type is **PF** (Physical file).
- ▶ Type in code in the LPEX editor.
- ▶ Press F4 in edit area for Source Prompter view.
- ▶ ...
- ▶ Compile
- ▶ Enter data in Client Access:
UPDDTA ACCOUNTS
- ▶ Query data:
RUNQRY *N ACCOUNTS

Homework?

- Finish Lab 1

Thank You!