**How To Run Chapter 3 Example Code**

**PsGpaRequest.ps1**

1. This is a powershell script to demo creating and using the PsGpaRequest cmdlet.
2. Navigate to \Code\powershell
3. In Windows Tools, open Windows PowerShell ISE
4. Open PsGpaRequest.ps1
5. Alter the -PrinterName or the -AttributesRequired variables to suit
6. Run the script (F5) or green arrow.

**GetPrinterAttributesRequest**

1. Navigate to \code\GetPrinterAttributesRequest\bin\Release
2. Usage: GetPrinterAttributesRequest /p=<printer\_name> /r=<requested\_attributes
3. Example: GetPrinterAttributesRequest.exe /p=HPM528 /r=printer-description

**IppPrinterQuery**

1. Navigate to \code\IppPrinterQuery\bin\Release
2. Note: An example printer attributes xml file (RequestedAttributes.xml) is included the project folder to use.
3. Usage: Usage: IppPrinterQuery /p=<list of printers file> /f=<attributes xml file>
4. Example: IppPrinterQuery.exe" /p=C:\Temp\PrinterList.txt /f= C:\Temp\ RequestedAttributes.xml

**WinIppTest**

Note: All the projects that use the LibCups Windows port wrapper should be opened in Visual Studio using Chap3\code\LibCups\vcnet\LibCups.sln. This is the C/C++ driver code for the wrapper. The wrapper is a demonstration project (WinIpp) that wraps some of the functionality of the CUPS C++ port.

1. Navigate to: code\LibCups\vcnet\x64\Release
2. Open a Console Window.
3. Run utility without any arguments to view help file – some example

**To print a pdf file to a printer:**

Usage: WinIppTest /p=printer\_url /d=pdf\_to\_print

Prints document on printer. Note: Only pdf/txt files supported.

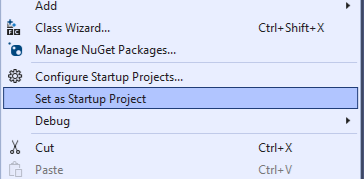
**Create a Job, and retrieves the job Id – asks to sends document to printer.**

Useage: WinIppTest /p=printer\_url /b=pdf\_to\_print

Example: WinIppTest.exe" /p=HPM528 /b=C:\temp\test.pdf

Note: Only pdf / txt files supported.

3. To modify code and debug, ensure you set it as the Startup Project.



**CS\_GPA\_CH3**

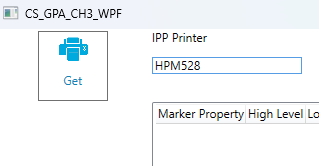
This is a C# utility that uses the WinIpp Wrapper of the CUPS C++ port. This utility demonstrates get-printer-attributes using .Net pInvoke functionality.

1. Navigate to \Code\LibCups\CS\_GPA\_CH3\bin\x64\Release
2. Execute CS\_GPA\_CH3.exe on a command line – no argument required.
3. The utility will ask for the target printer to use.
4. To modify the code or debug, ensure you set it as the Startup Project.

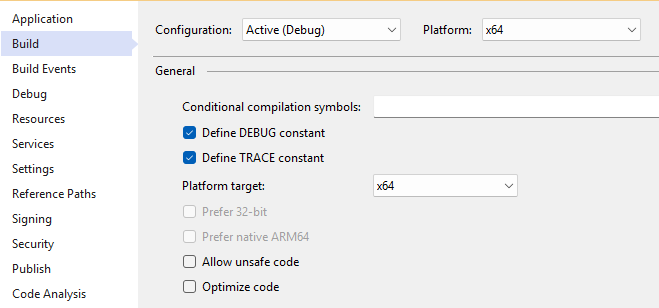
**CS\_GPA\_CH3\_WPF**

This is a utility uses the WinIpp Wrapper of the CUPS C++ port. This utility demonstrates get-printer-attributes using .Net pInvoke to display printer marker levels in a WPF GUI.

1. Navigate to \Code\LibCups\CS\_GPA\_CH3\_WPF\bin\x64\Release
2. Execute CS\_CH3\_WPF executable – this is a GUI program, no console required.
3. Enter the target printer in the IPP Printer textbox and click Get.



1. To modify the code or debug, ensure you set it as the Startup Project.
2. Notice: The platform must be set to x64.



**CS\_Multi\_Check**

This is a utility uses the WinIpp Wrapper of the CUPS C++ port. This utility demonstrates get-printer-attributes using .Net pInvoke to display attributes for multiple printers.

1. Navigate to: \code\LibCups\CS\_Multi\_Check\bin\x64\Release
2. Open a console Window.
3. Usage: CS\_Multi\_Check /l=<list of printers>
4. Example: CS\_Multi\_Check.exe" /l=C:\Temp\Printerlist.txt
5. To modify the code or debug, ensure you set it as the Startup Project.
6. Notice: The platform must be set to x64.