

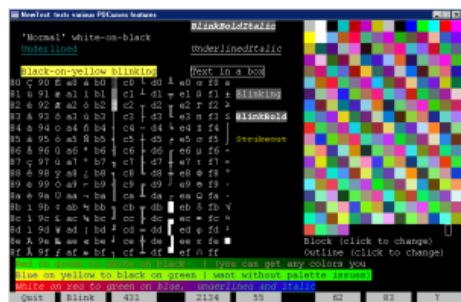
Adventures in C++ Programming.

How to set up PDCurses in Visual Studio 2013 (C++)

Posted in: Blog Post. Tagged: C++, How to, install, PDCurses, set up, Step by step, tutorial, Visual Studio 2013. 1 Comment

PDCourses

PDCurses is a public domain programming library compatible with Windows that provides an API for building “GUI-like” applications inside a console window. In a later post I will go into more detail on using PDCurses to build more robust console applications, but here is a taste of some of the functionality that PDCurses provides:



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/newtest.png>)

How to set up PDCurses in Visual Studio 2013

PDCurses is relatively easy to install. However, if you've never had to link to a library in Visual Studio 2013 it can be a pain, especially since recent changes to the latest version of VS have made a lot of the information found on the web obsolete. Luckily, I've provided a step by step guide to getting the PDCurses library up and running that applies to the latest version of Visual Studio including VS Express 2013.

Step 1) Download PDCourses

Download and unzip the following file:

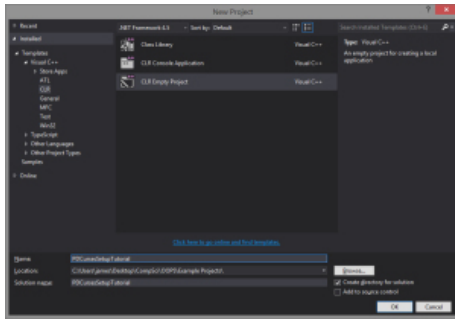
<http://www.filewatcher.com/m/pdc34dllw.zip.86939-0.html> (<http://www.filewatcher.com/m/pdc34dllw.zip.86939-0.html>)

Note: This is the only file you need to get started programming PDCurses applications. You do not need to find and download the entire library and compile it yourself.

After you've completed the download, unzip the file into a temporary directory on your desktop as we will be moving them into your project folder in a minute. You should have the following files all together: 'pdcurses.dll', 'pdcurses.lib', 'curses.h', and 'panel.h'.

Step 2) Create a Visual Studio project.

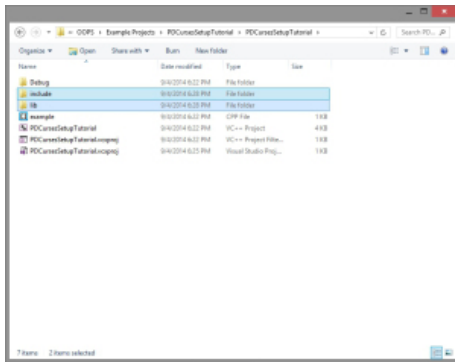
In this example, I'm using the following project settings:



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup1.jpg>)

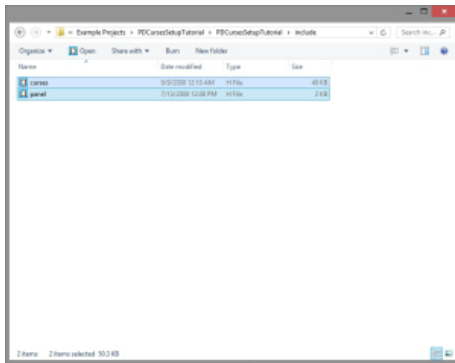
Step 3) Add PDCurses to your Visual Studio project folder.

In Windows, navigate to your project folder and create the folders 'lib' and 'include' as shown here:



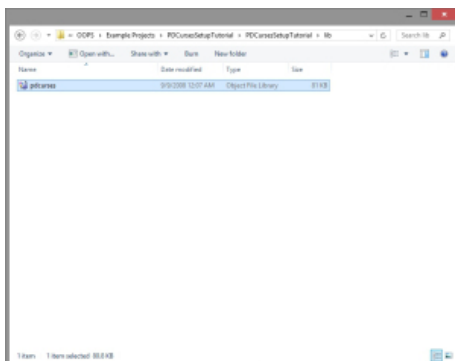
(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup31.jpg>)

Copy the header files 'curses.h' and 'panel.h' from the downloaded file into the include directory in your project like so:



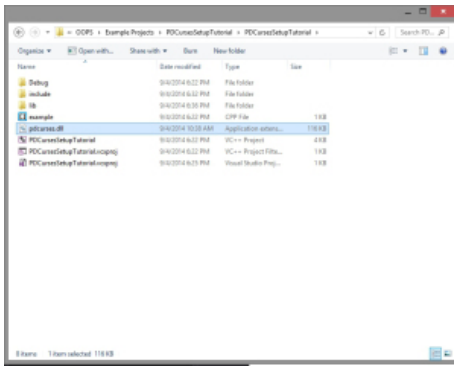
(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup4.jpg>)

Then, copy the file 'pdcurses.lib' from the downloaded file into the lib directory of your project folder:



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup5.jpg>)

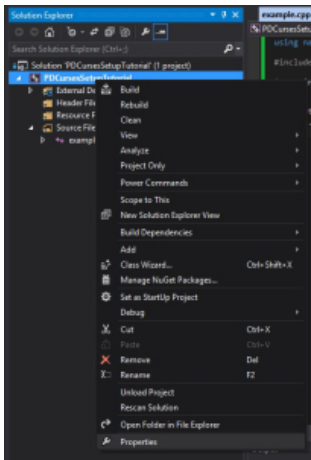
Finally, take 'pdcurses.dll' from the downloaded file and add it to the same folder as your VC++ Project file:



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup6.jpg>)

Step 4) Configure Visual Studio 2013

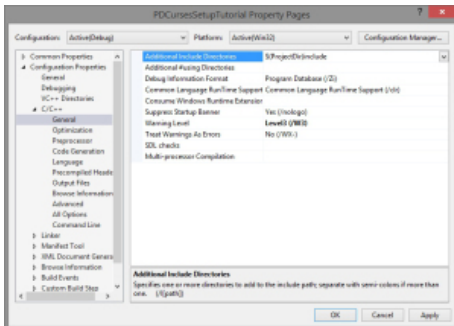
Right-click on the project file in the solution explorer and open the project properties page as shown:



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup2.jpg>)

Once you've opened the property pages for your project, on the left side under Configuration Properties, navigate to the C/C++ page and under the General tab add the following line in the box next to Additional Include Directories:

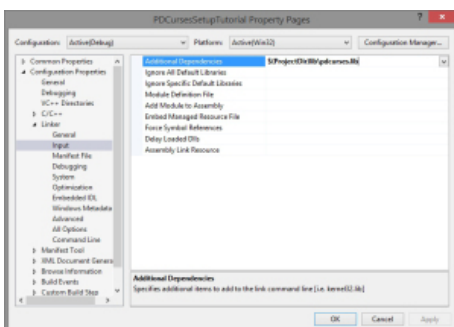
`$(ProjectDir)include`



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup7.jpg>)

Hit 'Apply' and then still under Configuration Properties, navigate to the Linker page, click on the Input tab and in the box next to Additional Dependencies, add the following line:

`$(ProjectDir)lib\pdcurses.lib`



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup8.jpg>)

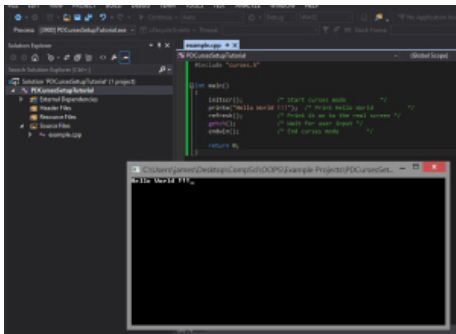
Step 6) You're all set!

If you've followed these steps you should be able to begin using the PDCurses API in Visual Studio 2013. If everything is set up correctly, the following code should print "Hello World !!!" to the console.

```
#include "curses.h"

int main()
{
    initscr();          /* Start curses mode */
    printw("Hello World !!!"); /* Print Hello World */
    refresh();          /* Print it on to the real screen */
    getch();            /* Wait for user input */
    endwin();           /* End curses mode */

    return 0;
}
```



(<https://jdonaldmccarthy.files.wordpress.com/2014/09/pdcsetup9.jpg>)

Documentation for the PDCurses API can be found at this link:

<http://pdcurses.sourceforge.net/doc/PDCurses.txt> (<http://pdcurses.sourceforge.net/doc/PDCurses.txt>)

For an in-depth tutorial on PDCurses functions, you may find the following ncurses resource below quite useful. Just remember that ncurses utilizes C, not C++, so there will be some very minor differences. Also you will need to `#include "curses.h"`, not as is given in the examples.

<http://www.tldp.org/HOWTO/NCURSES-Programming-HOWTO/intro.html> (<http://www.tldp.org/HOWTO/NCURSES-Programming-HOWTO/intro.html>)

One comment on "How to set up PDCurses in Visual Studio 2013 (C++)"

[hadlock](#) on February 2, 2015 at 7:15 am said:

Ah, thank you sir, very straightforward, but definitely makes me appreciate the simplicity of the Arduino way of handling libraries (drag folder to `~/Documents/Arduino/Libraries/` folder, restart Arduino IDE, done). Much appreciated! Saved me about an hour of trial and error.

[Reply](#)

[Create a free website or blog at WordPress.com.](#) [The Parament Theme.](#)