Using FWTools on Windows

FWTools is an easy way to install GDAL and OGR on Windows. It comes with its own version of Python and a bunch of other goodies. You can write your Python scripts in any text editor you want, but you do need to make sure that you run your scripts using the Python installation from FWTools. You can do this by running your script from inside the FWTools Shell or by configuring your editor to use the correct environment variables and Python executable. I wouldn't suggest trying to do this with PythonWin, however.

Installing FWTools

To install FWTools, just run the installer downloaded from http://fwtools.maptools.org/. Different versions of FWTools can coexist on the same computer without a problem.

Running a script from the shell

To run a script from the FWTools Shell, type:

```
python <script_name> [<args>]
```

You must provide the full path to the script unless it is in the current directory. For example, this doesn't work because I didn't provide the full path to my script:

```
C:\Program Files\FWTools2.2.8>python rasterinfo.py bear_lake.img
python: can't open file 'rasterinfo.py'
```

It still doesn't work because I didn't provide the full path to the input file:

```
C:\Program Files\FWTools2.2.8>python z:/data/classes/python/rasterinfo.py bear_lake.img
ERROR 4: `bear_lake.img' does not exist in the file system,
and is not recognised as a supported dataset name.
```

Now it works:

```
C:\Program Files\FWTools2.2.8>python z:/data/classes/python/rasterinfo.py
z:/data/imagery/aster/bear_lake.img
z:/data/imagery/aster/bear_lake.img has 5033 rows, 5665 columns, and 3 bands
```

I can also change into the directory of the script and run it without the full script path:

```
C:\Program Files\FWTools2.2.8>z:
Z:\>cd data/classes/python
Z:\Data\Classes\Python>python rasterinfo.py z:/data/imagery/aster/bear_lake.img
z:/data/imagery/aster/bear_lake.img has 5033 rows, 5665 columns, and 3 bands
```

Running Python interactively in the shell

You can also run Python interactively from the FWTools Shell by typing `python`. This will work like the interactive window in PythonWin, except that the up and down arrow keys actually work!

Customizing the shell

I think the QuickEdit mode is extremely useful in any Windows shell, including the FWTools one. To turn this on, right-click on the FWTools Shell icon and choose Properties. In the Properties dialog, select the Options tab. Make sure the box for QuickEdit mode is checked. This will allow you to highlight text within the shell and right-click to copy it. If you right-click twice then it will paste the highlighted text where your cursor is. If no text is highlighted but there is text on the clipboard, then right-clicking once will past the clipboard text into your shell. If QuickEdit mode is not enabled then it is much harder to copy and paste inside a shell.

You can also change the default colors and window size in the Properties dialog. The window size (on the Layout tab) is especially useful because Windows won't let you change the width of a shell window in the usual way.