Great Websites to Keep Programming!

All of these are free favorites of Wendy and Ryan

General Python Tutorials:

http://www.tutorialspoint.com/python/ nice divisions of material with good sample code and thorough introduction to the Python API.

http://zetcode.com/lang/python/ a good website for small samples of Python code and nice expanations. Plus there are a lot of other tutorials in the same style.

http://learnpythonthehardway.org/book/ a more traditional and individual approach to learning, very good for starting your first project alongside the book. You learn to make a game by the end!

http://inventwithpython.com/chapters/ an introductory Python book, also a bit more traditional and individual, but very worth it! It's full of fun connections to real-world examples.

Building a User Interface:

http://zetcode.com/gui/pyqt4/ a nice tutorial that explains basic concepts that are shared by many user interface tools, such as layouts and widgets.

http://pybee.org/ Ryan recommends the library for building a graphical user interface (GUI), which is http://pybee.org/toga/ (this site is a collection of projects that can be used to help develop, debug and launch Python software, not just write user interfaces, but each tool is self contained and can be used on its own)

Working with Data from Outside Python:

http://www.learnpython.org/en/Serialization Serializing data (such as lists or dicts) is important for saving data outside your program (to a file) and for communicating with other programs (sending JSON data to an angularJS application). **CPickle is not recommended** because it isn't guaranteed to work between Python versions, and has several security problems. **Tip:** You can add "indent=2" to a json.dumps call and it will print the data in a more readable format.

http://docs.python-requests.org/en/latest/user/quickstart/ As you write more programs you'll find yourself needing data from other places. Scores from last night's game, a list of tweets, the content of an RSS feed, and more. Requests is a Python library to make HTTP requests and present the data.

Using Python for Research or Engineering:

A gallery of academic IPython Notebooks These are some examples of academic research performed in Python, and can help you learn to explore data and draw conclusions.

http://wiki.scipy.org/Numpy_Example_List This page will make you very happy!!! The functions available in numpy are very intuitive for scientific researchers and engineers.

Miscellaneous:

http://stackoverflow.com/ very popular Q and A forum for all kinds of programming language, good for solving very specific questions you have about why one piece of code isn't working

Miscellaneous (continued):

https://github.com/ a great site for any coder, good for checking out Open Source projects or managing your own code. Git is a version control tool (helps you save multiple versions of the same file at different points in time in case you want them later, all stored in a repository) and Github is a web service that uses git, and also provides issue tracking, wikis, and a free project web site.

https://c9.io/ a cloud platform for development with Python and several other languages. Easy to code remotely with a partner and save files. The first workspace is free and can be divided into folders for multiple projects.