

Learnins Python

[Cross-posted from ICanHazDataScience]

Bad news. You're probably going to have to learn to code. Whilst you can go a very long way with the tools available online, at some point you're going to have that "if I could just reformat that column and extract this information out of it" moment. Which, generally, either means coding or finding a coder happy to help you with the task (hackathons like [RHOK](#) are good places, and always looking for good problem statements; there are also many coding-for-good groups around that might help too).

Not so bad news if you're up for writing your own code. There is *lots* of help available online. The language you choose is up to you... many social-good systems are written in PHP, for example, many open data systems are in Python (and there are a lot of good data-wrangling libraries available in Python and many data science courses use it as their default language), and R (free) and Matlab (not so free) are good for handling large arrays of data too.

I personally write most of my code in Python. This might not be your choice once you look at the other languages available, but it works for me, so that's what I'm going to write about (interspersed with a little PHP and R where it's appropriate).

So how do you start? When someone sends you a file with an name like "thingy.py", how do you run it?

You have options here, depending on what you want to do (run a file or code your own?) how much time you want to put it (two hours, a week, two years), and what your learning style is (reading text, watching video, doing tests, having a tutor). Most of these options are currently available free.

Here are some of them:

- Reading:
 - [The Best Way to Learn Python](#). An ambitious title, but it does give you an easy route through all the "standard" python tutorials. And I wish I'd had this when I started learning Python myself. If you're not a programmer, this is a good place to start.
 - [Learn Python the Hard Way](#). Comprehensive, but as it says, it's hard, but it makes things easier in the end
 - [The Python Tutorial](#). From Python coders, a good place for a seasoned coder to start.
- Watching:
 - [NewBoston's Python videos](#)

- Doing:
 - <http://www.pyschools.com/> A course that grades you as you go.
 - [Codecademy's Python course](#). Type your code into a box and test it online.
- Taught:
 - [MIT's Intro to Computer Science](#). Free online MIT course (yes, really!)
- More:
 - [Python's list of python tutorials](#)
 - [5 Best Websites to Learn Python](#)

And if you get stuck at any point in Python, do what I do and search [Stack Overflow](#) for a similar problem (and answers). If you can't find a similar problem (and you probably will!), then ask a question - it's full of friendly coders who respond well to something new.

And when you start using Python, you'll probably want these too:

- Places to practice:
 - [Python Challenge](#)
 - [Project Euler](#)
 - [TopCoder challenges](#) (can submit in Python)
- References:
 - When you start python, type `help(xx)` to get information about what you can do with the variable called `xx`. This can be very helpful, especially if you don't have the Internet at the time...
 - Online book: [Think Python](#)
- Useful libraries for data wrangling:
 - `csv`, `xlrd`, `xlwt`, `nltk`, [BeautifulSoup](#), `Pandas`

Onwards!