Software Literacy Begins With Learning To Code

Dick Maybach, Brookdale Computer Users Group, NJ

If you're reading this, you probably work with computers almost every day, and you probably also find yourself repeating some tasks many times, not because you enjoy them but because they have to be done. Often, you could reduce the tedium by writing a simple program.

For example, I belong to a club that uses its website to manage registrations for its activities, and we wanted to make available to the members a list of those registered for each event. The registration feature can download its data in the form of an Excel spreadsheet, but it includes data that we didn't want to make available on our site. It's of course possible to use Excel commands to extract and organize the information, but we often have several changes each day. Here's what has to be done.

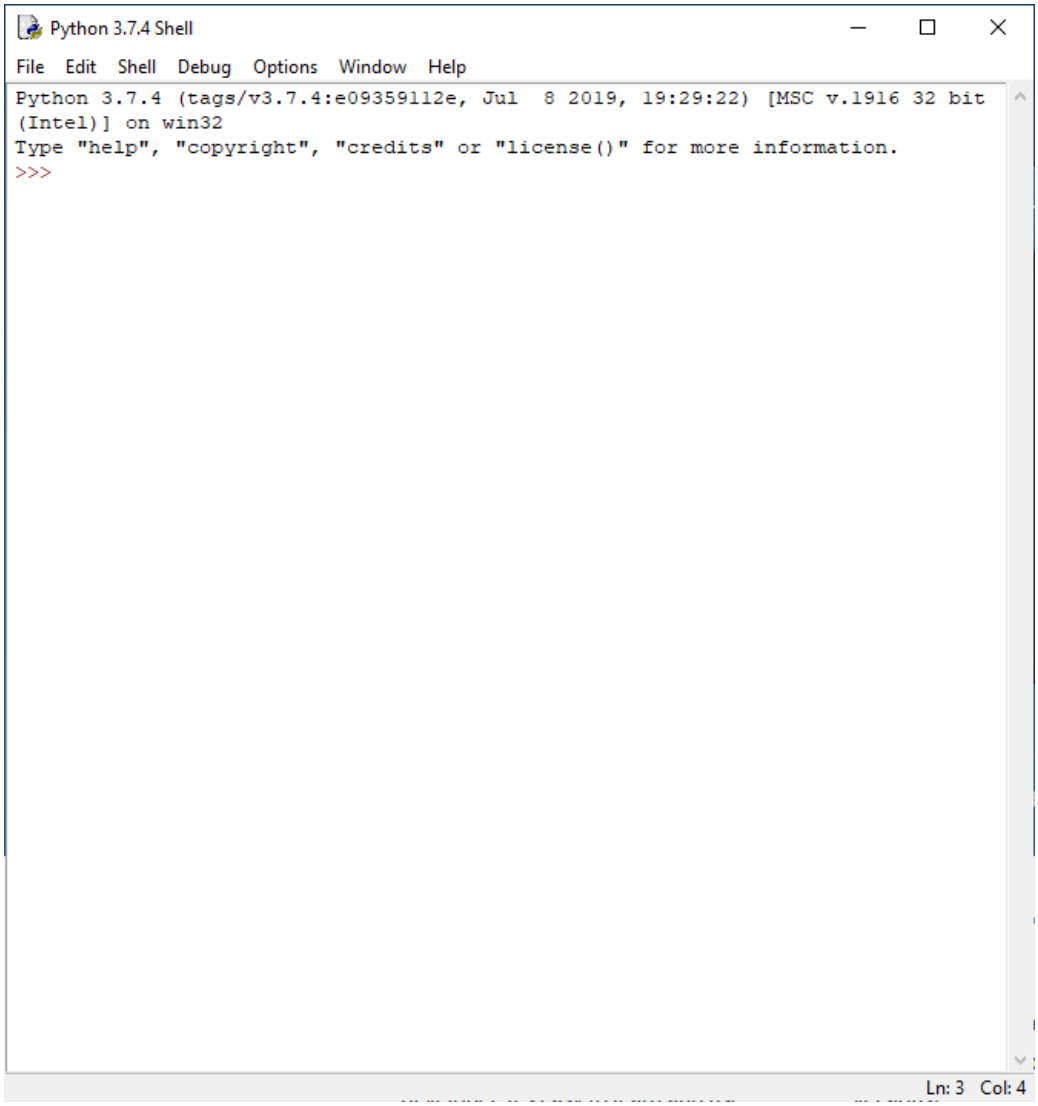
- Download the spreadsheet.
- Extract the first names (from column 4) and the last names (from column 5).
- Create a list with the format (last name, first name).
- Sort the list and format it with one item per line.
- Add HTML code so that it displays properly on a Web page.
- Paste the result onto our site.

I wrote a 13-line Python program that reduced the steps to these.

- Download the spreadsheet to my PC.
- Call the program with the spreadsheet file as an argument, which performs the above steps and places the result on my PC clipboard.
- Open the website and paste.

The program makes keeping the rosters updated much easier and with a greatly reduced chance of error.

I chose Python for this task, because it has extensions to read and write to applications such as Excel, and Word, download data from websites, schedule tasks, and send e-mail, see https://docs.python.org/3/py-modindex.html. That there are hundreds of applications is not an unmixed blessing, as many are evolving rapidly and keeping up can be challenging. You'll be making some Internet searches to find the current information. There are many tutorial books and articles on Python, for example see https://www.python.org/. For me, Automate the Boring Stuff with Python, by Al Sweigart, provided a good introduction to writing programs that access and manipulate data in office applications. However, Python Crash Course by Eric Matthes is a better introduction to the language with applications in computer games, data plotting, and Web page design.



As brick-and-mortar bookstores close and reduce their inventories, it becomes more difficult to find good programming books. If I can't find what I need locally, I make an Internet search on, for example, "Python Tutorial Books," and I usually find some good guidance on purchases and free books I can download. I've found computer-language reviews in magazines less helpful, as their space is too limited for adequate coverage.

That Python was a good choice for me at this time, of course says nothing about what language might be best for you. Learning one takes time and effort, and you want to take some care in your choice from the scores of those available. Some factors that might influence you are these.

- What operating systems and hardware support it?
- What are the costs of a development environment (editor, compiler, debugger, etc.)?
- Are tutorials, references, and application articles available?
- Is there an active user community?
- Is it currently under active development?

You may find better information in a group or publication interested in your application than one concerned with computers in general. For example, if you want to work with smart phone software, you might look to an Android interest group. I favor open-source approaches, as they reduce the costs.

Things may not work out, and that's OK. I once spent several weeks learning the Forth programming language, and my conclusion was, "I never want to do that again." I don't regret the effort, as learning is always worthwhile. If we never learn, we just repeat our daily lives, and there is a big difference between 12 years of experience and one year of experience repeated 12 times.