Lecture Notes Week 2

Welcome to Week 2. This week we will pick up the pace a bit and start digging into some very useful Python modules from the Standard Library. Becoming proficient in the standard libraries given to you by the programming language you choose (in this case the Python Standard Library) is a large part of becoming a good programmer. The idea for this week is to go over *a lot* of useful modules, but to not be too difficult either. If you find yourself struggling, please speak up, either via the discussion boards or email.

The modules that we will cover include datetime, logging, urllib, random, argparse and pprint. The readings for this week will have you learn what the modules do for you and how to use them. Also, since each of these modules is in the Python Standard Library, they all have documentation pages, which will be listed in the 'Useful Links' section under this week.

Datetime Module

The first module you will read about is the datetime module. Whenever you need to process date values, you should think of this module. It provides convenient data types when you are dealing with dates and times. With this module, you will be able to create variables that represent different datetimes, compare them (to see which date is later than another) and even do date arithmetic (to calculate how many days between two datetimes).

Logging Module

This next module is a module that I hope you will start using as a best practice. The logging module will allow you to record messages from your program by setup custom 'loggers'. For example, you can send all of your program's 'debug' output to a log file, but send any messages about errors to the screen. The module has a lot of features that will become useful to you as you become professional programmers.

urllib Module

If your program ever needs to download a file from a web or FTP server, the urllib module can help you do that pretty easily. Even though the module is pretty straight forward, there are some other libraries that can access resources from the internet, like the 'requests' module. However, urllib is the best tool for the job that comes with the Python standard library.

Random Module

The random module though simple, hides a lot of complexity. The random module is what we call a 'pseudorandom number generator', which allows programmers to use random numbers in their programs. You may be asking, 'what does pseudorandom mean'? Since computers, at their heart, are 'deterministic', there is nothing random about the instructions a computer can carry out. Computers, therefore, have to mimic randomness via algorithms that have been developed by computer scientists. This is why we call them pseudorandom.

Despite the name, however, these generated numbers are random enough for most purposes. The random module 'encapsulates' one of these algorithms, so you don't have to worry about the inner details of how these pseudorandom numbers are generated.

Argparse Module

Out of the modules we are learning about this week, argparse is the most complicated one. To help explain what it does, lets review how command line programs work.

When you run a program from the command line, you typically just use the name of the program. For example, when using Linux, you can run the 'ls' command to see a listing of files in your current directory by simply typing it into the shell:

$ ls

However, you can also pass 'parameters' to the program, to tell the program to run in a certain way. If I want to see detail about the files in my current directory, I can pass the l parameter (lowercase L):

$ ls -l

This is what is known as an option, or a flag. Some programs also accept what are known as positional arguments. For example, in Linux, to move a file from one folder to another, you can run:

$ mv file new_file

In this case, the parameters are passed by the position they are in. The first parameter must be the file you want to move, and the second parameter must be path of where it should go. Argparse allows you to write programs that can take parameters like shown above, making it easy for you to define an 'interface' to others using your program.

Pprint Module

We'll end with the pprint module, which is a very simple one: its sole purpose is to print out dictionaries, lists and other data types in a nice, readable way. Pprint is very useful when trying to debug your code, and you need to see the contents of a large dictionary that would otherwise be unreadable.

This Week's Takeaway

Bjarne Stroustrup, the creator of another programming language called C++, once said "The standard library saves programmers from having to reinvent the wheel". He was not talking about the Python Standard Library, but he could have. The deeper lesson is the same across all languages: Do not reinvent the wheel, unless you have to.