**Python Keywords and Functions**

The Python interpreter only knows a small number of words. They are:

**and** del from **not** **while**

as **elif** global **or** with

assert **else** **if** pass yield

**break** **except**  import **print**

class exec in raise

**continue** finally is return

def **for** lambda **try**

The list is small! Python is a ***dynamic*** ***language***, which means that this list will grow slowly as new versions of Python are developed. Words will never be taken away so as to keep the language backwards compatible.

The keywords highlighted in blue are words we have seen so far – about half of them!

Most programming languages only keep a small number of keywords like this. Having so few words seems like a bad idea, but in fact having few keywords means few restrictions on how things are done, and less complicated rules of syntax.

The power of a language like Python, despite the small language set, is that Python programmers have created thousands of functions that are available for everyone. A ***function*** is a short (or sometimes long) bit of code that can be called upon to do a specific job, such as draw a circle, take a square root or play an MP3 file. Once someone has created a function to do a task and posts it on the web, anyone can insert it into their own program. We will see how to use other people's functions and even create functions in the near future.

Functions are often grouped together into a ***module***. A module is like a package of useful functions for a particular task. For example, there is no keyword in Python for taking the square root of a number. So, someone wrote a function that does this. It is bundled in the **math** module. Here is how you would use it:

import math # look up the math module

x = 10

print (math.sqrt(x)) # print the square root of x

Notice that the sqrt function is prefixed with “math.”. This means “the sqrt function can be found in the math module”. It seems like overkill, but it saves the interpreter time if we can specify which module it can find our function. Also, as our programs get more sophisticated, there is a possibility that two modules share functions of the same name but different purpose.

Python is an ***open source language***, which means that the language itself and all the modules are available to everyone for free, and the code behind the modules is open for tinkering in case you can do a better job. There are modules available for math functions, graphical functions for animation, GUI functions to make windows apps, database functions (SQL), network client-server functions, webpage functions for HTML, XML and so on. The list is large and it grows every day with programmers around the world building new modules and functions.

Before we can use some of these functions that are available to us, we need to learn more about them. We will look next at how to make some simple functions of our own in the next lesson.

Key words: ***dynamic language, function, module, open source,***