Reviewing Python
Saturday Academy 8629, Spring, 2008
Instructor: Kirby Urner, 4D Solutions

Python is a computer language invented by Guido van Rossum and further enhanced by many talented individuals. Originally implemented in the C language, Python has since been migrated to the Java and .NET platforms.

In this class, we have looked at "three levels" of Python: data structures (like lists), functions (like in high school algebra), and classes (blueprints for objects).

Part of our fun has involved importing "namespaces" and doing computer graphics. This gave me an excuse to talk about two rather different ways to make computer animations (moving graphics) and to show movies (*Code Warrior* and *Warriors of the Net*). The two rather different ways: rendering and real time.

For a rendering approach, we looked at POV-Ray, a free ray-tracer (also open source). In this way of making a move (like *Cars*, *Over the Hedge* or *King Kong*), each frame of film takes a long time, so you need a "render farm" (a whole lot of computers) to meet deadlines.

For a real time approach, we used VPython (Visual Python) from Carnegie Mellon University, which bills itself as "3D programming for ordinary mortals." Computer games continually recompute the scene, the players, in real time, and so can't afford the luxury of rendering every detail so meticulously. With advances in chip technology, including at our local Intel, the real time animations are getting closer to rendered in quality.

Python helped us in doing both kinds of animation by supplying a lot of behind-the-scenes "scaffolding." Using polyhedra for objects, we could have Python either describe them in POV-Ray's scene description language, or we could invoke VPython directly, through its Pythonic API (API means Application Programming Interface, a way to "talk to" a resource, such as a web server or robot, from within a computer language).

Another key topic was ASCII becoming Latin-1 within Unicode. This is important in the Python world in that starting with Python version 3, rolling out this year, source code files default to Unicode's UTF-8, making it more like Java in this way. Coders will be able to use native character sets more easily, although Guido is planning to keep the Standard Library (the "batteries included" that come in the download) in mostly English, except for the author's name. Third party modules will not be bound by these restrictions.

Given this was a first exposure to Python, the emphasis was on overview and reading already-written code. If you want to practice on your own to further develop your skills, you'll find many excellent tutorials on the Web, or check the Saturday Academy course catalog for more options. Maybe we'll cross paths again, best wishes.