BATKY HOWELL

Python I: Essentials

This 4 day course leads the student from the basics of writing and running Python scripts to more advanced features such as file operations, regular expressions, working with binary data, and using the extensive functionality of Python modules. Extra emphasis is placed on features unique to Python, such as tuples, array slices, and output formatting. This is a hands-on programming class. All concepts are reinforced by informal practice during the lecture followed by graduated lab exercises. ***Python Programming*** is a practical introduction to a working programming language, not an academic overview of syntax and grammar. Students will immediately be able to use Python to complete tasks in the real world.

Course Objectives:

- Create working Python scripts following best practices
- Use python data types appropriately
- Read and write files with both text and binary data
- Search and replace text with regular expressions
- Get familiar with the standard library and its work-saving modules
- Know when to use collections such as lists, dictionaries, and sets
- Understand Pythonic features such as comprehensions and iterators

Audience: Users who want to learn Python and use it for application development, system administration, or just to automate tasks in a simple, yet powerful way.

Prerequisites: Students should already have a working, user-level knowledge of Unix/Linux, Mac, or Windows. While not required, basic skills with at least one other programming language will be helpful.

Number of Days: 4 days

1 Python Overview

What is Python?
The Birth of Python
Python Timeline
About Interpreted Languages
Advantages/Disadvantages of Python
How to get Python
Which version of Python?
The of 2.x
Getting help
pydoc

2 The Python Environment

Starting Python
If the interpreter is not in your PATHs
Using the Interpreter
Running a Python Script
Python Scripts on UNIX/Windows
Python Editors and IDEs

3 Getting Started

Using Variables
Keywords
Built-in Functions
Strings
Single-quoted string literals
Tripe-quoted string literals
Raw String literals
Unicode literals
String operators and expressions
Numeric literals
Math operators and expressions
Converting among types
Writing to the screen
String formatting
Legacy string formatting
Command line parameters
Reading from the keyboard