

This 4 day course picks up where Python I leaves off, covering some topics in more detail, and adding many new ones, with a focus on enterprise development. This is a hands-on programming class. All concepts are reinforced by informal practice during the lecture followed by lab exercises. Many labs build on earlier labs, which help students retain the earlier material.

Course Objectives:

- Leverage OS Services
- Code graphical interfaces for applications
- Create modules
- Create and run unit tests
- Define classes
- Interact with network services
- Query databases
- Process XML data

Audience: Advanced users, system administrators and web site administrators who want to use Python to support their server installations, as well as anyone else who wants to automate or simplify common tasks with the use of Python scripts.

Prerequisites: All students should be able to write simple Python scripts, using basic data types, program structures, and the standard Python library.

Number of Days: 4 days

1	Python Refresher Variables Python Data Types Sequences Mapping types Program Structure Files and console I/O Conditionals Loops Defining a Function Function Parameters Built-ins		The Time Module The Calendar Module
2	OS Services The <code>os</code> module Environment Variables Launching External Programs Paths, directories, and filenames Walking Directory Trees Dates and Times	3	Pythonic Programming Common Python Idioms Slicing and Dicing Unpacking Function Arguments Lambda Functions Nested Functions List Comprehensions Iterables Generator Expressions Writing Generators Python Time Travel Three Python Easter Eggs A String Trick String Formatting
		4	Modules and Packages Modules Using <code>import</code>

	Initialization Code		
	Namespaces		
	Executing modules as scripts		
	Packages		
	Configuring Import with <code>__init__.py</code>		
	Name Resolution (AKA scope)		
	Nested Functions		
	Python Style		
5	Classes	8	Distributing Modules
	About OO Programming		Installing Packages
	Defining Classes		Ways to distribute Code
	Constructors		Overview of distutils
	Instance Methods		Preparing for distribution
	Properties		Creating a Source Distribution
	Class Methods and Data		Creating built distributions
	Static Methods		setup.py options
	“Private” Methods		setup.py commands
	Inheritance	9	Database Access
	Untangling the Nomenclature		The DB API
6	Metaprogramming		Available Interfaces
	Special Attributes		Connecting to a Server
	<code>globals()</code> and <code>locals()</code>		Creating a Cursor
	Working with Attributes		Executing a Statement
	The <i>inspect</i> Module		Fetching Data
	Decorators		Parameterized Statements
	Decorator Functions		Dictionary Cursors
	Decorator Classes		Metadata
	Decorating Classes		Transactions
	Creating Classes at Runtime		Object-relational mappers
	Monkey Patching	10qt	GUI Programming with PyQt4
7	Objectives		What is PyQt4?
	Program Development		Event Driven Applications
	Comments		GUI Application Flow Chart
	pylint		External Anatomy of a PyQt4 Application
	Customizing pylint		Internal Anatomy of a pYQt4 Application
	Using pyreverse		Using designer
	The unittest module		Anatomy of a designer-based application
	Skipping Tests		Naming Conventions
	Making a suite of tests		Common Widgets
	Automated test discovery		Layouts
	Using Nose		Selectable Buttons
	The Python debugger		Making your application stretch
	Starting debug mode		Actions and Events
	Stepping through a Program		Menu bar
	Setting Breakpoints		Status bar
	Profiling		Using predefined dialogs
			Creating custom dialogs
			Tabs
			Niceties
			Working with Images
		10tk	GUI Programming with Tkinter
			Tkinter Overview

	Basic Tkinter Programming		Alternatives to Multiprogramming
	Object-oriented Tkinter	13	System Administration
	Widgets		The subprocess module
	Labels		subprocess convenience functions
	Buttons		Using the sh module
	Setting Fonts		Remote access
	Colors		Other useful modules
	Standard Colors		Permissions
	Variable Wrappers		Saving Information
	Selectable Buttons		Creating a useful command line scripts
	Text entry blanks		Creating Filters
	Multiline text entry/display		Parsing the command line
	Listbox		Simple logging
	Arranging Widgets		Logging Levels
	Using pack()		Formatting Log Entries
	Tweaking the layout		Logging to other Destinations
	Frames	14	XML and JSON
	Adding Scrollbars		About XML
	Callbacks		Normal approaches to XML
	Callback Parameters		Which module to use?
	Binding Events		Getting Started with ElementTree
	Event Specifications		How ElementTree works
	Creating Menus		Creating a new XML Document
11	Network Programming		Parsing an XML Document
	Sockets		Navigating the XML Document
	Socket Options		Using XPath
	Client Concepts		About JSON
	Server Concepts		Reading JSON
	Application Protocols		Writing JSON
	Forking Servers	15	Extending Python
	Grabbing HTML from the Web		Why extend Python?
	Consuming Web Services		Ways to extend Python with C
	Web Data the Easier Way		Hand-coded C
	Sending Email		Overview
	Binary Data		The C Program
	The struct Module		Methods
12	Multiprogramming		The Method Table
	What are Threads?		The init function
	The Python Thread Manager		Handling errors
	The threading Module		Custom exception objects
	Threads for the impatient		Putting it all together
	Creating a Thread Class		Using SWIG
	Variable Sharing		The interface file
	Using Queues		Generating the Wrappers
	Debugging Threaded Programs		Building and installing the extension
	The Multiprocessing Module		cytypes