

The following is a complete listing of class topics, taken from the Table of Contents of the class workbook. Please note that some topics covered are not included in this list, and class topics are prone to change over time. Please inquire about specific items.

Preface

About this class
Daily schedule
Course topics

1: General Python Introduction

So what's Python?
Why do people use Python?
Some quotable quotes
A Python history lesson
Advocacy News
What's Python good for?
What's Python not good for?
The compulsory features list
Python portability
On apples and oranges
Summary: Why Python?

Python Fundamentals

2. Using the Interpreter

Program execution model
Program architecture: modules
How to run Python programs
Configuration details
Module files: a first look
The IDLE interface
Other python IDEs
Time to start coding
Lab session 1

3. Types and Operators

Core datatypes introduction
Dynamic typing
Numbers
Strings
Lists
Dictionaries
Tuples
Files
General object properties
Summary: Python's type hierarchies
Built-in type gotchas

Lab session 2

4. Basic Statements

General syntax model
Assignment
Expressions
Print
If selections
Python syntax rules
Pydoc and documentation strings
Truth tests
While loops
Break, continue, pass, and the loop else
For loops
List comprehensions
Loop coding techniques
Comprehensive examples: file scanners
Basic coding gotchas
Preview: program unit statements
Lab session 3

5. Functions

Function basics
Scope rules in functions
More on “global”
More on “return”
Argument passing
Special argument matching modes
Demo: minimum value functions
Odds and ends
Design concepts: globals, accessors, closures
Functions are objects: indirect calls
Function gotchas
Optional case study set functions
Lab session 4

6. Modules

Module basics
Module files are a namespace
Import variants
Reloading modules
Package imports
__name__ and __main__
Odds and ends
Module design concepts
Modules are objects: metaprograms
Module gotchas
optional Case study: a shared stack module
Lab session 5

7. Classes

OOP: The big picture
Python class basics
Demo: People classes database
Using the class statement
Using class methods
Customization via inheritance
Specializing inherited methods
Operator overloading in classes
Namespace rules: the whole story
Design: inheritance and composition
Classes are objects: factories
Methods are objects: bound or unbound
Odds and ends
Class gotchas
optional Case study: a set class
Summary: OOP in Python
Lab session 6

8. Exceptions

Exception basics
First examples
Exception idioms
Exception catching modes
Matching variations
Exception gotchas
Lab session 7

9. Built-in Tools Overview

Debugging options
Inspecting name-spaces
Dynamic coding tools
Timing and profiling Python programs
Packaging programs for delivery
Summary: Python tool-set layers
Lab session 7

Python Applications

10. System Interfaces

System Modules overview
Arguments, Streams, Shell variables
File tools
Directory tools
Demo: finding large files
Forking processes
Thread modules and queues
The subprocess and multiprocessing modules [new]
IPC tools: pipes, sockets, signals [new]
fork versus spawnv
Demo: regression testing
Advanced system examples

Lab session 8

11. GUI Programming

Python GUI Options
The Tkinter 'hello world' program
Adding buttons, frames, and callbacks
Getting input from a user
Layout details
Demo: a Python/Tkinter GUI
Building GUIs by subclassing frames
Reusing GUIs by subclassing and attaching
Advanced widgets: images, grids, and more
Sexier examples
Tkinter odds and ends
Lab session 8

12. Databases and Persistence

Object persistence: shelves
Storing class instances
Pickling objects without shelves
Using simple dbm files
Shelve gotchas
Python SQL database API
ZODB object-oriented database
Demo: using MySQL from Python
Persistence odds and ends
Lab session 9

13. Text Processing

String objects: review
Splitting and joining strings
Demo: parsing data files
Regular expressions
Parsing languages
XML parsing: regex, Sax, DOM, and etree [new]
Lab session 10

14. Internet Scripting

Using sockets in Python
The FTP module
email processing
Other client-side tools
Writing server-side CGI scripts
Demo: an interactive Web Site in Python
Jython: Python for Java systems
Active Scripting and com
Python web frameworks
Other Internet-related tools
Lab session 10

15. Extending Python in C/C++

- Review: Python tool-set layers
- Stuff Guido already wrote
- Why integration?
- Integration modes
- A simple C extension module
- C module structure
- Binding C extensions to Python
- Data conversions: Python to/from C
- C extension types
- Using C extension types in Python
- Wrapping C extensions in Python
- Writing extensions in C++
- SWIG example
- Compiling with distutils
- Other extending options
- Python and rapid development
- Lab session 11

16. Embedding Python in C/C++

- General embedding concepts
- Running simple code strings
- Calling objects and methods
- Running strings: results & name-spaces
- Other code string possibilities
- Registering Python objects and strings
- Accessing C variables in Python
- C API equivalents in Python
- Running code files from C
- Precompiling strings into byte-code
- Embedding under C++
- More on object reference counts
- Integration error handling
- Automated integration tools
- Lab session 12

17. Advanced Topics [new]

- Unicode text and binary data
- Managed attributes
- Decorators
- Metaclasses
- Context managers
- Python 3.0 changes
- Lab session 13

18. Resources

- Python portability
- Major python packages
- Internet resources
- Python books
- Python in the news: articles, chapters

Python conferences and services
And finally

Classroom Exercises

Laboratory Exercises

Lab 1: Using the interpreter
Lab 2: Types and operators
Lab 3: Basic statements
Lab 4: Functions
Lab 5: Modules
Lab 6: Classes
Lab 7: Exceptions and built-in tools
Lab 8: System interfaces and GUIs
Lab 9: Persistence
Lab 10: Text processing and the Internet
Lab 11: Extending Python in C/C++
Lab 12: Embedding Python in C/C++
Lab 13: Decorators and metaclasses

Selected Exercise Solutions

Lab 1: Using the Interpreter
Lab 2: Types and Operators
Lab 3: Basic Statements
Lab 4: Functions
Lab 5: Modules
Lab 6: Classes
Lab 7: Exceptions and built-in tools
