python

Since 1999

BASCOM BRIDGE™

4th Floor Himalaya Complex, Opp. Sardar Centre, Nehru Park Fountain,
Vastrapur, Ahmedabad 380015, Gujarat, India.

URL: www.bascombridge.com

Help Line: +91 9376007676, E-mail: ranjana@bascombridge.com

Branches: Gandhinagar, Maninagar, VV Nagar

Advanced Python Programming

Bascom Bridge's Advanced Python Programming training course picks up where Introduction to Python Programming left off, covering some topics in more detail and adding new ones. For instance, classes are covered in greater detail, with new coverage of OS services, date/time management, binary data, unit testing, database connectivity, network programming, and more.

Prerequisite: Introduction to Python Programming

Duration: 30 Hours

ADVANCED PYTHON TRAINING OBJECTIVES

All students will:

- Leverage OS services
- Define classes
- Code graphical interfaces for applications
- Create modules and packages
- Implement and run unit tests
- Created multithreaded applications
- Interact with network services
- Query databases
- Process XML, CSV, and JSON data

ADVANCED PYTHON TRAINING OUTLINE

Introduction

Python refresher

- Data types
- Sequences
- Mapping types
- Program structure
- Files and console I/O
- Conditionals
- Loops
- Built-ins

OS Services

- The OS module
- Environment variables
- Launching external commands
- Walking directory trees
- Paths, directories, and filenames
- Working with file systems

Dates and times

- Basic date and time classes
- Different time formats
- Converting between formats
- Formatting dates and times
- Parsing date/time information

Binary Data

- What is Binary Data?
- Binary vs text
- Using the Struct module

Pythonic Programming

- The Zen of Python

- Common idioms
- Named tuples
- Useful types from collections
- Sorting
- Lambda functions
- List comprehensions
- Generator expressions

Functions, Modules, and packages

- Creating functions
- Function Parameters
- Variable scope
- Creating modules
- Using the import statement
- Module search path (PYTHONPATH)
- Documenting modules

Intermediate Classes

- Classes refresher
- Properties
- Class data/methods
- Inheritance
- Special methods

Metaprogramming

- Implicit properties
- globals() and locals()
- Working with object attributes
- The inspect module
- Callable classes
- Decorators
- Monkey patching

Developer Tools

- Analyzing programs with pylint
- Using the debugger
- Profilng code
- Testing speed with benchmarking

Unit tests

- What is a unit test
- Creating a Test cases
- Writing tests
- Running tests

Database access

- The DB API
- Available Interfaces
- Connecting to a server
- Creating and executing a cursor
- Fetching data
- Parameterized statements
- Using Metadata
- Transaction control
- ORMs and NoSQL overview

PyQt

- Overview
- Qt Architecture
- Using designer
- Standard widgets

- Event handling
- Extras

Network Programming

- Built-in classes
- Using requests
- Grabbing web pages
- Sending email
- Working with binary data
- Consuming RESTful services
- Remote access (SSH)

Multiprogramming

- The threading module
- Sharing variables
- The queue module
- The multiprocessing module
- Creating pools
- About async programming

Scripting for System Administration

- Running external programs
- Parsing arguments
- Creating filters to read text files
- Logging

Serializing data

- Working with XML
- XML modules in Python
- Getting started with ElementTree
- Parsing XML
- Updating an XML tree
- Creating a new document
- About JSON
- Reading JSON
- Writing JSON
- Reading/writing CSV files

Conclusion