

Advanced Python Programming

Duration: 24 Hours (3 Days)

Overview

The Advanced Python Programming course is designed for individuals looking to delve deeper into Python's powerful features and modules. This comprehensive course covers a wide range of topics from a Python refresher on basic concepts to Working with databases, Network programming, and Extending Python with C extensions. Through interactive lessons and hands-on exercises, learners will master Python's capabilities in analytics and data manipulation, making it an excellent python analytics course. Upon completion, participants will be equipped with the knowledge to tackle complex programming challenges and will be well-prepared to earn a python data analytics certification, showcasing their expertise in the field. By exploring advanced topics such as Metaprogramming, Multiprocessing, and Network programming, learners will enhance their skill set, preparing them for advanced Python applications in various industries.

Audience Profile

The Advanced Python Programming course by Koenig Solutions is designed for experienced programmers looking to deepen their understanding of Python and its applications.

- Software Developers seeking to improve Python coding efficiency
- Python Programmers aiming to learn advanced features and best practices
- Data Scientists requiring advanced Python skills for data manipulation and analysis
- DevOps Engineers interested in automating tasks and using Python for system management
- System Administrators who wish to script complex tasks in Python
- Backend Developers needing to master Python for web application development
- Full-stack Developers wanting to enhance their Python skills for versatility
- Software Engineers preparing for roles requiring in-depth knowledge of Python
- Technical Leads overseeing Python projects and seeking advanced knowledge
- IT Professionals wanting to transition into roles that require advanced Python expertise
- QA Engineers and Testers looking to automate testing with Python advanced features
- Data Analysts needing to perform data wrangling and analysis with Python
- Machine Learning Engineers who use Python for developing algorithms and models

Course Syllabus

Module 1 -- Python refresher

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

Module 2 -- OS Services

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

Module 3 -- Pythonic Programming

- The Zen of Python
- Common idioms
- Lambda functions
- List comprehensions
- Generator expressions
- String formatting

Module 4 – Modules and packages

- Initialization code
- Namespaces
- Executing modules as scripts
- Documentation
- Packages and name resolution
- Naming conventions
- Using imports

Module 5 -- Classes

- Defining classes
- Instance methods and data
- Properties
- Initializers
- Class and static methods/data
- Inheritance

Module 6 -- Metaprogramming

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

Module 7 – Programmer tools

- Analyzing programs
- Using pylint
- Testing code
- Using unittest
- Debugging

- Profiling and benchmarking



Module 8 -- Database access

- The DB API
- Available Interfaces
- Connecting to a server
- Creating and executing a cursor
- Fetching data
- Parameterized statements
- Metadata
- Transaction control
- Other DBMS modules

Module 9 -- Network Programming

- Sockets
- Clients
- Servers
- Application protocols
- Forking servers
- Binary data

Module 10 -- Multiprogramming

- When to use threads?
- The Global Interpreter Lock
- The threading module
- Simple threading
- Sharing variables
- The queue module
- Debugging threaded programs
- Multiprocessing
- Other alternatives

Module 11 – XML and JSON

- Working with XML
- DOM and Sax
- Introducing ElementTree and xml
- Parsing XML
- Navigating the document
- Creating a new XML document
- JSON
- Parsing JSON into Python
- Converting Python into JSON

Module 12 -- Extending Python

- About non-Python modules
- Overview of a C extension
- Writing C by hand

- Using SWIG
- Loading modules with ctypes

