

Certified Professional Python Programmer Level 2 (PCPP2)

Duration: 3 Days

Target Audience:

The PCPP2 course is designed for experienced Python programmers aiming to deepen their software development and design skills.

- Intermediate to advanced Python developers
- Software engineers
- DevOps engineers
- System architects
- Technical leads and team leaders
- Application developers
- Backend developers
- Software design engineers
- Full-stack developers
- IT professionals looking to learn about design patterns and package distribution
- Programmers interested in parallel programming and network programming
- Developers aiming to specialize in database programming with Python and MySQL
- Individuals preparing for the PCPP2 certification exam

Prerequisites:

To ensure that you are well-prepared for the PCPP2 – Certified Professional in Python Programming 2 course and can fully benefit from the training, the following are the minimum required prerequisites:

- Proficiency in Python programming: You should be comfortable with Python syntax and have experience writing Python code.
- Understanding of basic Python concepts: You should be familiar with concepts like data types, control flow, functions, classes, and exception handling.
- Familiarity with the Python Standard Library: Knowledge of commonly used modules in the Python Standard Library will be helpful.
- Experience with object-oriented programming (OOP): You should understand OOP principles such as inheritance, encapsulation, and polymorphism.
- Basic knowledge of software development tools: Comfort with using command-line interfaces, text editors, and version control systems (like Git) is recommended.
- Prior exposure to Python packaging: Some understanding of creating and managing Python packages, including the use of pip, would be beneficial.
- Basic understanding of relational databases: An introductory level of knowledge on how relational databases work, including SQL, is necessary for the Python-MySQL Database Access module.
- A willingness to learn and apply new concepts: As this is an advanced course, being open to learning about design patterns, interprocess communication, and network programming is important.

While not mandatory, prior completion of the PCPP1 – Certified Professional in Python Programming 1 or equivalent experience is highly recommended to ensure a smooth learning curve in this advanced course.

Remember, these prerequisites are intended to set you up for success in the PCPP2 course, and our goal at Koenig Solutions is to provide you with the skills and knowledge you need to advance your career in Python programming.

Course Objectives:

- Master the use of pip for package management and learn to structure, document, license, and distribute Python packages.
- Understand the setup.py file for package configuration and distribution to PyPI or other repositories.
- Acquire knowledge on various testing principles and techniques to ensure package reliability and integrity.

- Learn the core object-oriented design principles and how to implement design patterns such as Singleton, Factory, Façade, Proxy, Observer, Command, Template Method, MVC, and State.
- Develop the ability to write Python code that facilitates communication between processes using multiprocessing, threading, and subprocess modules.
- Master Python's socket module for network programming, enabling the development of network applications.
- Gain proficiency in relational database concepts and perform CRUD operations using Python with MySQL.
- Understand and differentiate between various database management systems, with a focus on MySQL.
- Learn to synchronize concurrent operations in a multiprocessing environment to manage shared resources.
- Develop the ability to architect and design robust, scalable, and maintainable applications by effectively applying design patterns and interprocess communication techniques.

Course Outlines:

Module 1: Creating and Distributing Packages

- Using pip
- Basic directory structure
- The setup.py file
- Sharing, storing, and installing packages
- Documentation
- License
- Testing principles and techniques
 - unittest – Unit testing framework
 - Pytest – framework to write tests

Module 2: Design Patterns

- Object-oriented design principles and the concept of design patterns
- The Singleton Design Pattern
- The Factory Pattern
- The Façade Pattern
- The Proxy Pattern
- The Observer Pattern
- The Command Pattern
- The Template Method Pattern
- Model-View-Controller
- The State Design Pattern

Module 3: Interprocess Communication

- multiprocessing — Process-based parallelism
- threading — Thread-based parallelism
- subprocess — Subprocess management
- Multiprocessing synchronisation
 - queue — A synchronized queue class
 - socket — Low-level networking interface
 - mmap — Memory-mapped file support

Module 4: Python Network Programming

- Python Socket Module
 - Introduction to sockets
 - Server Socket Methods
 - Client socket methods
 - General socket methods
 - Client-Server vs. Peer-to-peer
 - Other Internet nodules

Module 5: Python-MySQL Database Access

- Relational databases – fundamental principles and how to work with them
- MySQL vs. rest of the world
- CRUD Application

REGISTER NOW!

training@trends.com.ph
 (+632) 8863-2123
 www.trendscademy.com.ph