API Design with Python

Course Duration: 40 Hours (5 Days)

Overview

The API Design with Python course is a comprehensive program tailored for learners looking to gain expertise in building robust APIs using Python. This course covers a plethora of topics starting from a Python Review to ensure a strong foundation in Python's core concepts such as lists, dictionaries, and functions. As learners progress, they will delve into OS Interfacing with Python, mastering file and directory manipulation, and executing shell commands programmatically. The course then advances into Web and RESTful APIs, where students will learn the principles of REST, how to create Python clients for APIs, and manage authentication with tokens. The Python Protocol Clients module teaches scripting with HTTP and building clients and servers, while the JSON, YAML, XML, CSV, and Excel module focuses on data handling. Learners will also explore Generating and Sending Emails, handling Dates and Times, and utilizing Python Regular Expression (Regex) for advanced data processing. Code Review practices and tools like pylint ensure code quality, and Web API Design with Flask module provides hands-on experience in creating APIs with Python's popular micro-framework. The course also touches on database interactions through SQLite and concludes with Processes and Threads to understand concurrent programming in Python. By the end of this course, participants will be well-equipped to design, implement, and document APIs effectively, making them valuable assets in the tech industry.

Audience Profile

The "API Design with Python" course equips learners with essential skills for developing robust APIs using Python, catering to IT professionals looking to enhance their expertise.

- Software Developers and Engineers
- Backend Developers
- Systems Architects
- Data Scientists who need to integrate systems or expose data through APIs
- DevOps Engineers involved in continuous integration and deployment processes
- IT Professionals focusing on web services and cloud solutions
- Full Stack Developers looking to strengthen their backend skills
- Technical Project Managers overseeing API development projects
- Technical Leads and Senior Developers designing system architectures
- Quality Assurance Engineers specializing in API testing
- Application Developers building APIs for mobile or desktop applications
- Network Engineers interested in network automation using Python APIs
- Security Professionals assessing API security and conducting penetration testing

Course Syllabus

KOENIG
step forward

1. Python Review

- Lists
- Dictionaries
- Tuples
- Conditionals (if, elif, else)
- Loops (for and while)
- Functions
- Variable scope review
- Converting boilerplate code to functional code
- Writing reusable functions
- Using pip
- Useful third-party modules
- Publishing a module
- Documenting modules

2. OS Interfacing

- os module: listdir(), getcwd(), mkdir()
- Common sys module attributes & methods
- Working with the os module and files/directories
- Walking file trees with os.walk()
- File, path, and directory operations
- Working with os.path
- os and os.path module examples
- Environment variables
- Running shell commands
- Compressing and archiving (gzip, tar, zip)

3. Web and RESTful APIs

- Understanding REST
- REST APIs and HTTP CRUD operations
- REST and OpenStack
- URI analysis and formation
- Capturing requests with Wireshark
- Using cURL
- etcd keystore
- Creating a Python client to interact with API endpoints
- API development keys

- SSH and Python
- Secure password retrieval
- Tokens and APIs

4. Python Protocol Clients

- Scripting the browser
- Scripting with HTTP
- Creating an HTTP client and server
- Python and SSH
- Building an SFTP client and server
- Python and SFTP limitations
- Using Paramiko for SSH
- Netmiko for major network vendors (Cisco, Juniper, Arista)

5. JSON, YAML, XML, CSV, and Excel

- JSON (RFC 7159)
- JSON formatting
- Introduction to YAML
- YAML lists and dictionaries
- YAML line spanning
- Reading YAML files
- XML parsing
- Working with CSV files
- Importing JSON and YAML
- Decoding JSON and YAML in Python
- Using Python to process data structures (YAML, XML, CSV, JSON)
- Reading from Excel files
- Writing to Excel files

6. Generating and Sending Emails

- Overview of email modules
- Creating simple emails
- Interfacing with email accounts

7. Dates and Times

- Python and cron
- Importing time and using time.time()
- Suspending execution with sleep()

- Working with time data
- Time formatting
- Time tuples
- Creating calendars

8. Python Regular Expressions (Regex)

- Metacharacter review
- re module basics
- search() and match() functions
- findall() method
- Compiling regex search patterns
- Creating efficient searches
- Sorting datasets
- sort() vs. sorted()
- Sorting with functions
- Applying regex to file searches
- Applying regex to API results

9. Code Review

- Best practices
- Using pylint for code analysis
- Python coding conventions
- Using underscores in variable names
- Double underscores (__) and name mangling
- Monkey patching

10. Web API Design with Flask

- Flask overview
- Using decorators
- Building APIs with Python and Flask
- APIs returning Jinja2 templating
- Returning a cookie
- Building sessions
- Redirecting from URIs
- Building an API to accept file uploads

11. SQLite

- Overview of SQLite

- Connecting SQLite with Python
- Read and write operations
- Useful SQLite commands
- Connecting APIs with SQLite
- Reading and writing to databases using APIs

12. Processes and Threads

- Threading basics
- Context switching
- Deadlock errors
- Thread starvation
- Race conditions and race specifics
- Working with locks