

55285A: Advanced Python

Duration: 16 Hours (2 Days)

Overview

The 55285A: Advanced Python course is a comprehensive training program designed for individuals looking to deepen their knowledge and expertise in advanced coding in Python. This course covers a range of sophisticated topics, from Lambda functions to Advanced list comprehensions and the Collections module in Module 1. Learners will explore the intricacies of data handling in Module 2, including Working with relational databases, CSV files, and Extracting data from the web. In Module 3, the focus shifts to Testing and debugging for optimal performance. Finally, Module 4 dives into Classes and objects, where Attributes, Inheritance, and Decorators are demystified. This course is regarded as the best course for advanced Python due to its thorough curriculum and practical applications, addressing complex topics that prepare learners for real-world programming challenges. It is tailored to help participants refine their Python capabilities, ensuring they can write more efficient, effective, and high-quality Python code.

Audience Profile

The 55285A: Advanced Python course is designed for experienced Python developers looking to deepen their knowledge in advanced programming, data handling, and software design.

- Target audience for the course includes:
- Software Developers and Programmers with Python experience
- Data Scientists and Analysts looking to enhance data manipulation skills
- Database Administrators interested in Python for automation and data processing
- Data Engineers who want to improve their data pipeline scripting abilities
- Backend Developers aiming to refine their coding and debugging practices
- DevOps Engineers who use Python for scripting and automation tasks
- IT Professionals needing advanced Python skills for system optimization
- Software Architects designing complex systems with Python
- Quality Assurance Engineers focused on test automation in Python
- Technical Leads and Managers overseeing Python projects
- Academic Researchers utilizing Python for complex data analysis

Course Syllabus

Module 1: Advanced Python Concepts

- In this lesson, you will learn about some Python functionality and techniques that are commonly used but require a solid foundation in Python to understand.

Lessons

- Lambda Functions
- Advanced List Comprehensions
- Collections Module

- Mapping and Filtering
- Mutable and Immutable Built-in Objects
- Sorting
- Unpacking Sequences in Function Calls

Lab: Exercises in this Lesson

- Rolling Five Dice
- Creating a defaulted
- Creating a OrderedDict
- Creating a Counter
- Working with a deque
- Converting list.sort() to sorted(iterable)
- Converting a String to a datetime.date Object
- After completing this module, students will be able to:
- Work with lambda functions.
- Write more advanced list comprehensions.
- Work with the collections module to create named tuples, defaultdicts, ordereddicts, counters, deque
- Use mapping and filtering.
- Sort sequences.
- Unpack sequences in function calls.
- Create modules and packages.

Module 2: Working with Data

- Data is stored in many different places and in many different ways. There are Python modules for
- all of the most common ways.

Lessons

- Relational Databases
- CSV
- Getting Data from the Web
- JSON

Lab: Exercises in this Lesson

- Querying a SQLite Database
- Inserting File Data into a Database
- Comparing Data in a CSV File
- Requests and BeautifulSoup
- Using JSON to Print Course Data
- After completing this module, students will be able to:
- Access and work with data stored in a relational database.
- Access and work with data stored in a CSV file.
- Get data from a web page.

- Access and work with data stored as HTML and XML.
- Access an API.
- Access and work with data stored as JSON.

Module 3: Testing and Debugging

- This module explains how to test and debug using Python.

Lessons

- Testing for Performance
- The unittest Module

Lab: Exercises in this Lesson

- Fixing Functions
- After completing this module, students will be able to:
- Test performance with timers and using the timeit module.
- To write unit tests using the unittest module.

Module 4: Classes and Objects

- An object is something that has attributes and/or behaviors, meaning it is certain ways and does
- certain things. In the real world, everything could be considered an object. Some objects are
- tangible, like rocks, trees, tennis racquets, and tennis players. And some objects are intangible,
- like words, colors, tennis swings, and tennis matches.

Lessons

- Attributes
- Behaviors
- Classes vs. Objects
- Attributes and Methods
- Private Attributes
- Properties
- Documenting Classes
- Inheritance
- Static Methods
- Class Attributes and Methods
- Abstract Classes and Methods
- Understanding Decorators

Lab: Exercises in this Lesson

- Adding a roll() Method to Die
- Properties
- Documenting the Die Class
- Extending to Die Class
- Extending the roll() Method
- After completing this module, students will be able to:

- Create classes and objects in Python.
- Write instance methods, class methods, and static methods.
- Define properties.
- Create subclasses using inheritance.
- Create abstract classes.
- Appropriately document Python classes.
- Understand how decorators work.

Additional Reading

- None