

Course Outline

Introduction to Python Programming | Python Programming Basics Course TTPS4800: 3 days Instructor Led

About this course

Introduction to Python | Python Programming Basics is a hands-on Python programming course that teaches you the key skills you'll need to get started with programming in Python to a solid foundational level. The start of the course will lead you through writing and running basic Python scripts, and then guide you through how to use more advanced features such as file operations, regular expressions, working with binary data, and using the extensive functionality of Python modules. Extra emphasis is placed on features unique to Python, such as tuples, array slices, and output formatting.

This course provides you with an excellent kick start for users new to Python and scripting, enabling you to quickly use basic Python skills on the job in a variety of ways. You'll be able use Python in basic web development projects or use it to automate or simplify common tasks with the use of Python scripts. The course also serves as a solid primer course / foundation for continued Python study in support for next level web development with Python, using Python in DevOps, Python for data science / machine learning or Python for systems admin or networking support.

Audience profile

This basic level course provides an excellent kick start for users new to Python and scripting, enabling them to use basic Python skills on the job in a variety of ways. This course is appropriate for advanced users, system administrators and web site administrators who want to use Python to support their server installations, as well as anyone else who wants to automate or simplify common tasks with the use of Python scripts.

Students can apply the course skills to use Python in basic web development projects or automate or simplify common tasks with the use of Python scripts. The course also serves as a solid primer course / foundation for continued Python study in support for next level web development with Python, Python for data science / machine learning or Python for systems admin or networking support. Basic familiarity with any programming or scripting language would be helpful. Students should have a working, user-level knowledge of Unix/Linux, Mac, or Windows.

At course completion

After completing this course, students will be able to:

- Create working Python scripts following best practices
- Use python data types appropriately
- Read and write files with both text and binary data
- Search and replace text with regular expressions
- Get familiar with the standard library and its work-saving modules
- Create "real-world", professional Python applications
- Know when to use collections such as lists, dictionaries, and sets
- Understand Pythonic features such as comprehensions and iterators
- Write robust code using exception handling

Course Outline

1. An overview of Python
 - What is python?
 - Python Timeline
 - Advantages/Disadvantages of Python

Course Outline

- Getting help with pydoc
2. The Python Environment
 - Starting Python
 - Using the interpreter
 - Running a Python script
 - Python scripts on Unix/Windows
 - Editors and IDEs
 3. Getting Started
 - Using variables
 - Builtin functions
 - Strings
 - Numbers
 - Converting among types
 - Writing to the screen
 - Command line parameters
 4. Flow Control
 - About flow control
 - White space
 - Conditional expressions
 - Relational and Boolean operators
 - While loops
 - Alternate loop exits
 5. Array types
 - About array types (AKA sequences)
 - Lists and list methods
 - Tuples
 - Indexing and slicing
 - Iterating through a sequence
 - Nested sequences
 - Sequence functions, keywords, and operators
 - List comprehensions
 - Generator Expressions
 6. Working with files
 - File overview
 - Opening a text file
 - Reading a text file
 - Writing to a text file

Course Outline

7. Dictionaries and Sets

- About dictionaries
- Creating dictionaries
- Iterating through a dictionary
- About sets
- Creating sets
- Working with sets

8. Functions

- Defining functions
- Returning values
- Parameters
- Global and local scope

9. Sorting

- The **sorted()** function
- Alternate keys
- Lambda functions
- Sorting collections
- Using operator.itemgetter()
- Reverse sorting

Time-Permitting

10. Errors and Exception Handling

- Syntax errors
- Exceptions
- Using try/catch
- Handling multiple exceptions
- Ignoring exceptions

11. Modules and Packages

- The **import** statement
- Module search path
- Creating Modules
- Using packages
- Function and Module aliases

12. Working with Classes & OO Programming

- About o-o programming
- Defining classes
- Constructors

Course Outline

- Methods
- Instance data
- Properties
- Class methods and data