

Course duration

- 4 days

Course Benefits

- How Python works.
- Python's place in the world of programming languages.
- Python literals.
- Python comments.
- Variables and Python data types.
- Simple modules.
- Outputting data with print().
- Collecting user input.
- Defining and calling functions.
- Parameters and arguments.
- Variable scope.
- Creating and importing modules.
- Math in Python.
- The math and random modules.
- String basics.
- Special characters.
- Multi-line strings.
- Indexing and slicing strings.
- Common string operators and methods.
- Formatting strings.
- Built-in string functions.
- Lists, tuples, ranges, dictionaries, and sets.
- The *args and **kwargs parameters.
- Virtual environments.
- Installing packages with pip.
- Conditions and loops.
- Generator functions.
- List comprehensions.
- Exception handling.
- The time and datetime modules.
- Working with files and directories.
- Working with the os and os.path modules.
- PEP8 and Pylint.

Available Delivery Methods

Private Class

Private classes are delivered for groups at your offices or a location of your choice.

Self-Paced

Learn at your own pace with 24/7 access to an On-Demand course.

Course Outline

1. Python Basics
 1. Getting Familiar with the Terminal
 2. Running Python
 3. Running a Python File
 4. Exercise: Hello, world!
 5. Literals
 6. Exercise: Exploring Types
 7. Variables
 8. Exercise: A Simple Python Script
 9. Constants and Deleting Variables
 10. Writing a Python Module
 11. print() Function
 12. Collecting User Input
 13. Exercise: Hello, You!
 14. Reading from and Writing to Files
 15. Exercise: Working with Files
2. Functions and Modules
 1. Defining Functions
 2. Variable Scope
 3. Global Variables
 4. Function Parameters
 5. Exercise: A Function with Parameters
 6. Returning Values
 7. Exercise: Parameters with Default Values
 8. Returning Values
 9. Importing Modules
 10. Methods vs. Functions
3. Math
 1. Arithmetic Operators
 2. Exercise: Floor and Modulus
 3. Assignment Operators
 4. Precedence of Operations
 5. Built-in Math Functions
 6. The math Module
 7. The random Module

8. Exercise: How Many Pizzas Do We Need?
9. Exercise: Dice Rolling
4. Python Strings
 1. Quotation Marks and Special Characters
 2. String Indexing
 3. Exercise: Indexing Strings
 4. Slicing Strings
 5. Exercise: Slicing Strings
 6. Concatenation and Repetition
 7. Exercise: Repetition
 8. Combining Concatenation and Repetition
 9. Python Strings are Immutable
 10. Common String Methods
 11. String Formatting
 12. Exercise: Playing with Formatting
 13. Formatted String Literals (f-strings) (introduced in Python 3.6)
 14. Built-in String Functions
 15. Exercise: Outputting Tab-delimited Text
5. Iterables: Sequences, Dictionaries, and Sets
 1. Definitions
 2. Sequences
 3. Lists
 4. Sequences and Random
 5. Exercise: Remove and Return Random Element
 6. Tuples
 7. Ranges
 8. Converting Sequences to Lists
 9. Indexing
 10. Exercise: Simple Rock, Paper, Scissors Game
 11. Slicing
 12. Exercise: Slicing Sequences
 13. min(), max(), and sum()
 14. Converting between Sequences and Strings
 15. Unpacking Sequences
 16. Dictionaries
 17. The len() Function
 18. Exercise: Creating a Dictionary from User Input
 19. Sets
 20. *args and **kwargs
6. Virtual Environments, Packages, and pip
 1. Exercise: Creating, Activating, Deactivating, and Deleting a Virtual Environment
 2. Packages with pip
 3. Exercise: Working with a Virtual Environment
7. Flow Control
 1. Conditional Statements
 2. Compound Conditions
 3. The is and is not Operators

4. all() and any() and the Ternary Operator
5. In Between
6. Loops in Python
7. Exercise: All True and Any True
8. break and continue
9. Looping through Lines in a File
10. Exercise: Word Guessing Game
11. The else Clause in Loops
12. Exercise: for...else
13. The enumerate() Function
14. Generators
15. List Comprehensions
8. Exception Handling
 1. Exception Basics
 2. Generic Exceptions
 3. Exercise: Raising Exceptions
 4. The else and finally Clauses
 5. Using Exceptions for Flow Control
 6. Exercise: Running Sum
 7. Raising Your Own Exceptions
9. Python Dates and Times
 1. Understanding Time
 2. The time Module
 3. Time Structures
 4. Times as Strings
 5. Time and Formatted Strings
 6. Pausing Execution with time.sleep()
 7. The datetime Module
 8. datetime.datetime Objects
 9. Exercise: What Color Pants Should I Wear?
 10. datetime.timedelta Objects
 11. Exercise: Report on Departure Times
10. File Processing
 1. Opening Files
 2. Exercise: Finding Text in a File
 3. Writing to Files
 4. Exercise: Writing to Files
 5. Exercise: List Creator
 6. The os Module
 7. os.walk()
 8. The os.path Module
 9. A Better Way to Open Files
 10. Exercise: Comparing Lists
11. PEP8 and Pylint
 1. PEP8
 2. Pylint

Class Materials

Each student will receive a comprehensive set of materials, including course notes and all the class examples.

Class Prerequisites

Experience in the following *would be useful* for this Python class:

- Some programming experience.

Follow-on Courses

- [Advanced Python 3 Training](#)
- [Python Data Analysis with JupyterLab Training](#)