



Let's Reach For Excellence!

TAN DUC INFORMATION TECHNOLOGY SCHOOL JSC

Address: 103 Pasteur, Dist.1, HCMC
Tel: 08 38245819; 38239761
Email: traincert@tdt-tanduc.com
Website: www.tdt-tanduc.com; www.tanducits.com

A Practical Introduction to Python Programming

Duration: 04 days

Though Python® has been in use for nearly thirty years, it has become one of the most popular languages for software development, particularly within the fields of data science, machine learning, artificial intelligence, and web development. Whether you're relatively new to programming, or have experience in other programming languages, this course will provide you from beginning to Intermediate in Python.



COURSE OUTLINE

I. Basic

Chapter 1: Getting Started

1. Custom
2. Installing Python
3. IDLE
4. A first program
5. Typing things in
6. Getting input
7. Printing
8. Variables
9. Exercises

Chapter 2: For loops

1. Examples
2. The loop variable
3. The range function
4. A Trickier Example
5. Exercises

Chapter 3: Numbers

1. Integers and Decimal Numbers
2. Math Operators
3. Order of operations
4. Random numbers
5. Math functions
6. Getting help from Python
7. Using the Shell as a Calculator
8. Exercises

Chapter 4: If statements

1. A Simple Example
2. Conditional operators
3. Common Mistakes
4. elif
5. Exercises

Chapter 5: Miscellaneous Topics I

1. Counting
2. Summing
3. Swapping
4. Flag variables

5. Maxes and mins
6. Comments
7. Simple debugging
8. Example programs
9. Exercises

Chapter 6: Strings

1. Basics
2. Concatenation and repetition
3. The in operator
4. Indexing
5. Slices
6. Changing individual characters of a string
7. Looping
8. String methods
9. Escape characters
10. Examples
11. Exercises

Chapter 7: Lists

1. Basics
2. Similarities to strings
3. Built-in functions
4. List methods
5. Miscellaneous
6. Examples
7. Exercises

Chapter 8: More with Lists

1. Lists and the random module
2. split
3. join
4. List comprehensions
5. Using list comprehensions
6. Two-dimensional lists
7. Exercises

Chapter 9: While loops

1. Examples
2. Infinite loops
3. The break statement
4. The else statement
5. The guessing game, more nicely done

6. Exercises

Chapter 10: Miscellaneous Topics II

1. str, int, float, and list
2. Booleans
3. Shortcuts
4. Short-circuiting
5. Continuation
6. pass
7. String formatting
8. Nested loops
9. Exercises

Chapter 11: Dictionaries

1. Basics
2. Dictionary examples
3. Working with dictionaries
4. Counting words
5. Exercises

Chapter 12: Text Files

1. Reading from files
2. Writing to files
3. Examples
4. Wordplay
5. Exercises

Chapter 13: Functions

1. Basics
2. Arguments
3. Returning values
4. Default arguments and keyword arguments
5. Local variables
6. Exercises

Chapter 14: Object-Oriented Programming

1. Python is object-oriented
2. Creating your own classes
3. Inheritance
4. A playing-card example
5. A Tic-tac-toe example
6. Further topics
7. Exercises

II. Graphics

Chapter 15: GUI Programming with Tkinter

1. Basics
2. Labels
3. grid
4. Entry boxes
5. Buttons
6. Global variables
7. Tic-tac-toe

Chapter 16: GUI Programming II

1. Frames
2. Colors
3. Images
4. Canvases
5. Check buttons and Radio buttons
6. Text widget
7. Scale widget
8. GUI Events
9. Event examples

Chapter 17: GUI Programming III

1. Title bar
2. Disabling things
3. Getting the state of a widget
4. Message boxes
5. Destroying things
6. Updating
7. Dialogs
8. Menu bars
9. New windows
10. pack
11. StringVar
12. More with GUIs

Chapter 18: Further Graphical Programming

1. Python 2 vs Python 3
2. The Python Imaging Library
3. Pygame

III. Intermediate

Chapter 19: Miscellaneous topics III

1. Mutability and References
2. Tuples
3. Sets
4. Unicode
5. sorted
6. if-else operator
7. continue
8. eval and exec
9. enumerate and zip
10. copy
11. More with strings
12. Miscellaneous tips and tricks
13. Running your Python programs on other computers

Chapter 20: Useful modules

1. Importing modules
2. Dates and times
3. Working with files and directories
4. Running and quitting programs
5. Zip files
6. Getting files from the internet
7. Sound
8. Your own modules

Chapter 21: Regular expressions

1. Introduction
2. Syntax
3. Summary
4. Groups
5. Other functions
6. Examples

Chapter 22: Math

1. The math module
2. Scientific notation
3. Comparing floating point numbers
4. Fractions
5. The decimal module
6. Complex numbers
7. More with lists and arrays

8. Random numbers
9. Miscellaneous topics
10. Using the Python shell as a calculator

Chapter 23: Working with functions

1. First-class functions
2. Anonymous functions
3. Recursion
4. map, filter, reduce, and list comprehensions
5. The operator module
6. More about function arguments

Chapter 24: The itertools and collections modules

1. Permutations and combinations
2. Cartesian product
3. Grouping things
4. Miscellaneous things from itertools
5. Counting things
6. defaultdict

Chapter 25: Exceptions

1. Basics
2. Try/except/else
3. try/finally and with/as
4. More with exceptions