

Python Training Course

Beginner' s Course

Introduction to Python

Lesson 1

- Why Python?
Brief historical review of programming: compiled / interpreted languages
What motivated the invention of Python (Guido Van Rossum)
Success of Python: league table
- Jupyter Notebook programming environment.
Installation, features
- Python first steps: environment, libraries, packages
- Exercise: First Python Script

Lesson 2

- Python: Object Orientation I
rather than dive into syntax
script to demonstrate object programming
bring in syntax as required
- Function definitions – what objects do – object attributes
- Exercise: mammal catalogue for Casella

Lesson 3

- Objects are typed
- Inbuilt types
- Invented types: classes
- Text types: strings, bytes, Unicode
- Intro data structures
- Exercise: Tic Tac Toe

Lesson 4

- Python Data Structures I
- Sequences: tuples and lists, queues, stacks
- For loops
- List comprehensions
- Numpy arrays
- List exercises

Lesson 5

- Python Data Structures II
- Mapping Types: Dictionaries
- Keys, hashing
- Dictionary Views
- Ordering
- Dict Exercises

Lesson 6

- Python Data Structures III
- **With** statement
- Context Managers
- Managing resources
- Reading files
- Exercise: counting lines in a file

Lesson 7

- Syntax I
- Variables in depth, underscore
- Assignment
- Background memory management
- Int and float
- Exercise: Python Calculator

Lesson 8

- Syntax II
- Operators
- Basic types in depth
- Intro strings, bytes, UTF-8
- Print formatting
- Exercise: formatting text

Lesson 9

- Syntax III
- Functions, arguments
- Inbuilt
- Useful examples of inbuilt functions
- Functional programming
- Exercise: Textual Manipulation

Lesson 10

- Python: Object Orientation II
- Reviewing OOP
- Inheritance
- Class methods
- Class decorators
- Exercise: Casella revisited

Lesson 11

- Python libraries: sys, os
- Packages
- Package management
- Creating packages
- Installing packages

Lesson 12

- Managing Errors!
- Exceptions
- Inbuilt
- Creating exceptions
- Testing