

Chapter 1: Getting Started with Python

This chapter introduces readers to the basic syntax of the Python language, taking a practical approach. It begins by recommending the installation of the Anaconda distribution, which includes Python 3.7+, IPython, Spyder, NumPy, SciPy, and matplotlib. Spyder is recommended as a development environment due to its integrated interface.

The fundamental concepts covered include basic data types (numbers, strings, lists), control structures (loops, conditions), functions, scripts, and modules. Special attention is given to the use of the Python interpreter and the IPython environment.

The chapter concludes with the difference between scripts and modules, and how to import and execute them in interactive sessions.

Chapter 2: Variables and Basic Types

This chapter delves deeper into the fundamental types in Python: integers, floats, complex numbers, Booleans, and strings. It emphasizes that Python automatically deduces the type of variables, which are actually references to objects.

It explores the peculiarities of floats (representation, machine epsilon, inf, nan), complex numbers (form, real/imaginary part, conjugate method), and booleans (logical operators, implicit conversions). Practical examples show how types interact in conditional and mathematical expressions.

The chapter ends with an in-depth presentation of character strings, including escape sequences, operations (concatenation, repetition), common methods (split, join, find), and advanced formatting via `format()` and f-strings.

Chapter 3: Container Types

This chapter covers data structures for storing multiple elements: lists, tuples, dictionaries, sets, and arrays.

Lists are covered in detail: indexing, slicing, insertion, deletion, concatenation, iterations, list comprehensions, and built-in methods (append, extend, sort, etc.).

NumPy arrays are briefly introduced, with a warning about their fundamental differences from lists (type homogeneity, vector operations, shared views/memory).

Tuples, which are immutable structures, are presented along with the concepts of packing/unpacking. Dictionaries are explored from the perspective of key/value pairs, with creation, modification, and traversal. Sets allow you to perform set operations (union, intersection, etc.).

Finally, conversions between container types are summarized, as well as ways to check the type of a variable via ``type()`` and ``isinstance()``.