

About Python

Chapter 1

This Chapter covers

- Why use Python?
- What Python does well
- What Python doesn't do as well
- Why learn Python 3?

Why Use Python?

- Easy to learn and use
- Mature and supportive Python community
- Hundreds of Python libraries and frameworks
- Versatility, efficiency, reliability, and speed
- Big Data, Machine Language and Cloud Computing
- First-choice Language
- The flexibility of Python language
- Use of Python in academics
- Automation

Background

- Python is a modern programming language developed by Guido van Rossum in the 1990s (and named after a famous comedic troupe).
- Although Python isn't perfect for every application, its strengths make it a good choice for many situations.

What Python Does Well

- Python is easy to use
- Python is expressive
- Python is readable
- Python is complete – “batteries included”
- Python is cross-platform
- Python is free

What Python Doesn't Do Well

- Python isn't the fastest language
- Python doesn't have the most libraries
- Python doesn't check variable types at compile time
- Python doesn't have much mobile support
- Python doesn't use multiple processors well

Why Learn Python 3?

- Python has been around for a number of years and has evolved over that time.
- Python 3, originally whimsically dubbed Python 3000, is notable because it's the first version of Python in the history of the language to break backward compatibility.
- What this means is that code written for earlier versions of Python probably won't run on Python 3 without some changes.
- Why learn Python 3? Because it's the best Python so far.

Summary

- Python is a modern, high-level language with dynamic typing and simple, consistent syntax and semantics.
- Python is multiplatform, highly modular, and suited for both rapid development and large-scale programming.
- It's reasonably fast and can be easily extended with C or C++ modules for higher speeds.
- Python has built-in advanced features such as persistent object storage, advanced hash tables, expandable class syntax, and universal comparison functions.
- Python includes a wide range of libraries such as numeric processing, image manipulation, user interfaces, and web scripting.
- It's supported by a dynamic Python community.