

The Core Language

1 Introduction

1.1 Why Python?

2 How is Python Different?

2.1 Python Concepts

2.1.1 Dynamic vs. Static Types

2.1.2 Interpreted vs. Compiled

2.1.3 Prototyping

2.1.4 Procedural vs. Object-Oriented Programming

3 Comparison of Programming Languages

3.1 C

3.2 C++

3.3 Java

3.4 C#

3.5 Python

4 The Python Interpreter

4.1 Launching the Python interpreter

4.1.1 Windows

4.1.2 Mac

4.2 Python Versions

Using the Python Command Prompt

4.3 Commenting Python

4.4 Launching Python programs

4.5 Integrated Development Environments

5 Types and Operators

5.1 Python Syntax

5.1.1 Indentation

5.1.2 Multiple Line Spanning

5.2 Python Object Types

5.3 Python Numbers

6 Strings

6.1 Basic string operations

6.2 Indexing and slicing strings

6.3 String Formatting

6.4 Combining and Separating Strings

6.5 Regular Expressions

7 Lists

7.1 List usage

7.2 Adding List Elements

7.3 Mutability

7.4 Methods

8 Dictionaries

8.1 Making a dictionary

8.2 Basic operations

8.3 Dictionary details

8.4 Operation

9 Tuples

9.1 Why Use Tuples?

9.2 Sequence Unpacking

9.3 Methods

10 Files

10.1 File Operations

10.2 Files and Streams

10.3 Creating a File

10.4 Reading From a File

10.5 Iterating Through Files

10.6 Seeking

10.7 Serialization

11 Statements

11.1 Assignment

11.2 Expressions/Calls

11.3 Printing

11.4 *if* Tests

11.5 *while* Loops

11.6 *for* Loops

11.7 *pass* Statement

11.8 *break* and *continue* Statements

11.9 *try*, *except*, *finally* and *raise* Statements

11.10 *import* and *from* Statements

11.11 *def* and *return* Statements

11.12 Class Statements

12 Lambda Expression

12.1 Making Python Do Something

12.2 Creating Lambda Expression

12.3 Map, Filter and Reduce

12.4 Default Arguments



13 Exceptions

- 13.1 Exception Class Hierarchy
- 13.2 User-Defined Exceptions

14 Object Oriented Programming

- 14.1 Learning Python Classes
- 14.2 How Are Classes Better?
- 14.3 Improving Your Class Standing
- 14.4 So What Does a Class Look Like?
- 14.5 “New-style” classes
- 14.6 A Note About Style

15 More OOPS

- 15.1 Inheritance
- 15.2 Operator Overloads
- 15.3 Class Methods
- 15.4 Abstraction & Polymorphism

16 Databases

- 16.1 How to Use a Database
- 16.2 Working With a Database
- 16.3 Using SQL to Query a Database
- 16.4 Python and SQLite
- 16.5 Creating an SQLite DB
- 16.6 Pulling Data from a DB
- 16.7 SQLite Database Files

17 Threading

- 17.1 Creating Threads with Threading Module
- 17.2 Synchronizing Threads
- 17.3 Priority Queue

18 Networking

- 18.1 Programming Socket Programming
- 18.2 Application on Socket

19 Email Services

- 19.1 Sending Email
- 19.2 Sending Email with Attachments

20 Distributing Your Program

II Graphical User Interfaces

21 Overview of Graphical User Interfaces

- 21.1 Introduction
- 21.2 GUI Controls(Tkinter/Qt5)

22 Web Scripting

- HTML 5
- CSS3
- JAVA SCRIPT

23 Web Application

- 23.1 Django

24 Control Hardware Devices Using Python

- 24.1 Working on Sensors, Led, Motors etc.
- 24.2 Home Automation
- 24.3 Working With Sim900
- 24.4 Creating Various Application Using Python To Control Hardware

25 Raspberry Pi B+3

- 25.1 Introduction to Raspberry Pi
- 25.2 Installation & Setting up Pi
- 25.3 Setting up GPIO Pins
- 25.4 Serial Communication with Pi
- 25.5 Synchronous Peripheral Interfaces
- 25.6 Sensors Interfacing

26 IOT & Cloud Interfacing

- 26.1 Introduction to IOT
- 26.2 Introduction of Cloud Computing
- 26.3 AWS/PUBNUB/AZURE Cloud
- 26.4 Creating Dashboard on Cloud
- 26.5 Creating IOT Device on Cloud
- 26.6 Sending & Fetching Data from Cloud
- 26.7 M2M Communication Using Cloud

