# Basic Python

## A Beginner's Guide that
## Hopefully Makes Learning Python Fun and Easy

By

*Kellen Sun*

MIT
License

Copyright (c) 2020 kellen-sun

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN
NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE
SOFTWARE.

This book is dedicated to:

Every single person who has read it. And one man who will never read it Tony Stark.

# Table of Contents

# Chapter Zero: Introduction

In the modern world, computers are crucial to life. More and more schools teach programming and more and more employers hire people who can code. This book's purpose is to teach the reader python. You might be wondering what python is. Simply put python is a programming language. There are many other programming languages. Each one has its own advantages. This can be thought as why there are many human languages like English, French, Spanish to name a few. All these languages have different words and syntax, but they all do the same thing. They allow and simplify communication between humans. A programming language also does this. It allows and simplifies communication between humans and computers. Now why does this book teach python and not another programming language. The main advantages of python are listed below.

- ➢ Cross-platform (python can run on almost any computer)
- ➢ Beginner friendly
- ➢ Useful
- ➢ Powerful (python can be used for advanced topics)

I am writing this book to teach python and to make more people discover its beauty and simplicity, but I am also writing to improve my own skills because teaching something is a good way to see if you've learnt it well. Best of luck going forward.

## About Python

As we have already discussed, python is a programming language. It allows a human to communicate to a computer. Python comes in many different versions. It is recommended to always use the newest available version. Code in this book has been written in python 3.6.1. It should work on newer versions, and maybe even some older versions. The two main branches of python are python2 and python3. They are vastly different and even simple programs could fail to work on the other version. All work should be done in python3. This book teaches Python because it is popular, easy to learn, simple, powerful and free. The official python website is http://python.org/. You should visit for more information when needed. If you've picked up this book, you're probably excited to learn python or someone told you to read this book. Before we start learning python there are a few basic requirements that should be met.

## Requirements

Expected:

The reader is expected to know how to read.

The reader is expected to know how to operate a computer. This means opening and saving files and searching the web.

The reader is expected to have a computer because the book does not provide one.

Not Expected:

The reader is not expected to be a computer expert.

The reader is not expected to know how to code.

## Who is this book for?

Anyone can read this book. However, the book is recommended for absolute beginners who haven't written any code before. If you want to learn python of coding and have never started, this book is a great place to start. Other books might be better though. This book will attempt to keep everything simple while delivering all the necessary important information. I would recommend this book to teens between the ages of 10-18 years old. If you're like 60, you can still read it. It's okay, boomer.

## Conventions Used

This book will attempt to make every normal human being distinguish where there is code, text, links, comments, symbols and images. If difficulty arises, try your best or get another book. I'm sorry if the book isn't clear enough. Perhaps you come from Mars, I've always wanted to visit. Finally, if any problems arise throughout this book, you're welcome and expected to google your issues and to try and fix them yourself. Making errors and thinking and trying are the very best ways to learn python and anything.

## Getting Started

The first step to learning python is to have python installed on your computer. This requires installing python. To install python first figure out your operating system. If you have difficulties follow this short guide.

1. If your computer has an apple on it, the operating system is probably Mac OS X.

2. If in one of the corners of your screen your computer has a small window panel symbol of a start button, then your computer is probably a Windows.

3. If it's none of the above, it's probably a Linux computer.

Now that the operating system of your computer has been figured out follow these steps to install python. Examples in this book are provided using Windows 10, python 3.6.1, Geany and Google Chrome.

1. Mac OS X;

Python is normally already installed on most Mac computers. To verify it's installed open the terminal and type 'python'. If this doesn't raise an error then you should be good. Next install a text editor. If any errors occur refer to installation issues.

2. Windows

To install python on windows, open a web browser and go to the site http://python.org/downloads/. Download the newest one or the one with the highest number, it should be 3.something.something. Then follow the instructions to install. If any errors occur refer to installation issues.

3. Linux

Most Linux computers have python already installed to check this open a terminal and type Python. Make sure the version is up to date. If any errors occur refer to installation issues.

## Installation Issues

If you have any installation issues, you should google "install python on _____" with your operating system name. Then watch a video or choose a guide. If this still doesn't work, you can start by writing your code in an online text editor. Websites such as repl.it and onlinegdb.com work great. While you're at it also attempt to install pip or the python package installer.

# Chapter One: Absolute Basics

In this chapter, you'll hopefully learn to open, write, and save python programs.

In the world of computers there are 2 main things that a computer will care about. The first being data. Data is information. Data can be stored in many ways as texts or numbers. Those different ways are known as data types in python. And in python there are a few ways that the computer can manage it. The way a computer operates data is referred to as functions. Operating on data includes changing it, creating it, removing it, and performing calculations on it.

In this chapter we'll learn our first data type strings and our first function the print statement.

To create our first basic code, we'll need a text editor. Any editor will work and many are available. I recommend using something simple such as Geany. Instruction for installation can be found online. So, install any editor you want. Now if this doesn't work you can work with an online editor for now. Refer to installation issues. This book will assume you're using Geany. For fun you can go through Geany's settings and adapt it to your taste and preferences. Note: this book is not sponsored by Geany I am merely using it to show examples.

## Starting

So, open your editor. Find the 'New' button in the upper right or under 'file' -> 'new' or just press Ctrl+N. At this moment a new tab should appear. We'll then click 'file' -> 'save as'. Choose the directory or folder where you want to save all your python code for this book or create a new folder if necessary. Finally save the file as 'hello_world.py'. Notice, how the file ends in .py this ensures the file can be understood by python. It is a norm to use lowercases and underscores instead of spaces in filenames. I recommend you to do so also. However, I cannot force you to do it. After the python file is saved make sure your keyboard is working and get mentally and physically ready to write some not so epic code.

## Comments

The first thing we'll learn is comments because the are easy and make you look cool. Comments in python allow you and other people to understand the code better by commenting on it. Comments are useful for code created and shared between multiple people. Comments are created with a hashtag. When python sees a hashtag, it will ignore everything on the same line and continue reading your code on the next line. Comments can be used to explain assumptions, important decisions, problems and details about your code. Here is an example of a comment.

# Epilogue/Conclusion

Insert content here…

Keep it short and sweet. Mention the key highlights of the book and the action steps to solve the problems. Remind the reader of how taking the said action steps will benefit them.

# **Bibliography**

Below is an example of a list of works cited using APA style. Arrange your list of references alphabetically.

When citing books, use the format as shown in the following examples, applying Bibliography style to the format:

Author's last name, first initial or initials. (Publication date). *Book title*. Additional information. City, State of publication: Publishing company.

King, S. (2000). *On writing: A memoir of the craft.* New York, NY: Pocket Books.

When citing online resources, use the format as shown in the following examples:

For internet documents

Author's last name, first initial or initials. (Date of publication). Title of article. *Title of work*. Retrieved from full URL

Amir, N. (2018, October 17). 4 tips for staying on track with your writing. *Write Nonfiction now!* Retrieved from http://writenonfictionnow.com/tips-staying-track-writing/

For online periodicals

Author's last name, first initial or initials. (Date of publication). Title of article. *Title of Periodical*, volume and page numbers. Retrieved from full URL

Brewer, R. L. (2018, October 4). How to write better titles: 7 effective title tips for books, articles, and conference sessions. *Writer's Digest.* Retrieved from http://www.writersdigest.com/whats-new/how-to-write-better-titles

When citing magazines, use the format as shown in the following examples:

For magazines and periodicals

Author's last name, first initial or initials. (Publication date). Article title. *Title of periodical, volume number (issue number if available*), inclusive pages.

McPhee, J. (2013, April 29). Draft No. 4. *New Yorker*, *89*, 20-25.

For more details and guidelines, consult the APA Publication Manual.

# Acknowledgments

Insert content here…

Thank the key people who inspired you and helped you throughout the process of writing and publishing your work. This is somewhat similar to the dedication page, except here you can elaborate and include more people.

# About the Author

Insert content here…

Write this page in the third person. (For example, use "the author" or your name, not "I.") The information within establishes your credibility with readers. Avoid being too wordy. Simply provide your background and expertise on the topic of your book, as well as other information that will build trust. For example:

- Professional and personal achievements related to the topic at hand

- List of other published works and a link to your website

- Educational background

- Mention other notable experts in the field that you have worked with

- Your area of residence, family status, hobbies, etc.