

Scientific Computing With Python Notes

Patrick Adams

November 1, 2020

Note: This is a draft copy of notes generated by free code camp.
<https://www.freecodecamp.org/>

Contents

| | | |
|----------|---|----------|
| 1 | Python For Everybody | 4 |
| 1.1 | Introduction: Why Program? | 4 |
| 1.2 | Introduction: Hardware Architecture | 4 |
| 1.3 | Introduction: Python as a Language | 4 |
| 1.4 | Introduction: Elements of Python | 4 |
| 1.5 | Variables, Expressions, and Statements | 4 |
| 1.6 | Intermediate Expressions | 4 |
| 1.7 | Conditional Execution | 4 |
| 1.8 | More Conditional Structures | 4 |
| 1.9 | Python Functions | 4 |
| 1.10 | Build your own Functions | 4 |
| 1.11 | Loops and Iterations | 5 |
| 1.12 | Iterations: Definite Loops | 5 |
| 1.13 | Iterations: Loop Idioms | 5 |
| 1.14 | Iterations: More Patterns | 5 |
| 1.15 | Strings in Python | 5 |
| 1.16 | Intermediate Strings | 5 |
| 1.17 | Reading Files | 5 |
| 1.18 | Files as a Sequence | 5 |
| 1.19 | Python Lists | 5 |
| 1.20 | Working with Lists | 5 |
| 1.21 | Strings and Lists | 5 |
| 1.22 | Python Dictionaries | 5 |
| 1.23 | Dictionaries: Common Applications | 5 |
| 1.24 | Dictionaries and Loops | 5 |
| 1.25 | The Tuples Collection | 5 |
| 1.26 | Comparing and Sorting Tuples | 5 |
| 1.27 | Regular Expressions | 6 |
| 1.28 | Regular Expressions: Matching and Extracting Data | 6 |
| 1.29 | Regular Expressions: Practical Applications | 6 |
| 1.30 | Networking with Python | 6 |
| 1.31 | Networking Protocol | 6 |
| 1.32 | Networking: Write a Web Browser | 6 |
| 1.33 | Networking: Text Processing | 6 |
| 1.34 | Networking: Using urllib in Python | 6 |
| 1.35 | Networking: Web Scraping with Python | 6 |
| 1.36 | Using Web Services | 6 |
| 1.37 | Web Services: XML | 6 |
| 1.38 | Web Services: XML Schema | 6 |
| 1.39 | Web Services: JSON | 6 |
| 1.40 | Web Services: Service Oriented Approach | 6 |
| 1.41 | Web Services: APIs | 6 |
| 1.42 | Web Services: API Rate Limiting and Security | 6 |
| 1.43 | Python Objects | 7 |
| 1.44 | Objects: A Sample Class | 7 |
| 1.45 | Object Lifecycle | 7 |
| 1.46 | Objects: Inheritance | 7 |
| 1.47 | Relational Databases and SQLite | 7 |
| 1.48 | Make a Relational Database | 7 |
| 1.49 | Relational Database Design | 7 |
| 1.50 | Representing Relationships in a Relational Database | 7 |
| 1.51 | Relational Databases: Relationship Building | 7 |

| | | |
|----------|--|----------|
| 1.52 | Relational Databases: Join Operation | 7 |
| 1.53 | Relational Databases: Many-to-many Relationships | 7 |
| 1.54 | Visualizing Data with Python | 7 |
| 1.55 | Data Visualization: Page Rank | 7 |
| 1.56 | Data Visualization: Mailing Lists | 7 |
| 2 | Scientific Computing With Python Projects | 8 |
| 2.1 | Arithmetic Formatter | 8 |
| 2.2 | Time Calculator | 8 |
| 2.3 | Budget App | 8 |
| 2.4 | Polygon Area Calculator | 8 |
| 2.5 | Probability Calculator | 9 |

1 Python For Everybody

1.1 Introduction: Why Program?

More resources:

- Install Python on Windows
- Install Python on MacOS

1.2 Introduction: Hardware Architecture

1.3 Introduction: Python as a Language

1.4 Introduction: Elements of Python

1.5 Variables, Expressions, and Statements

1.6 Intermediate Expressions

More resources:

- Exercise 1
- Exercise 2

1.7 Conditional Execution

1.8 More Conditional Structures

More resources:

- Exercise 1
- Exercise 2

1.9 Python Functions

1.10 Build your own Functions

More resources:

- Exercise

1.11 Loops and Iterations

1.12 Iterations: Definite Loops

1.13 Iterations: Loop Idioms

1.14 Iterations: More Patterns

More resources:

- Exercise

1.15 Strings in Python

1.16 Intermediate Strings

More resources:

- Exercise

1.17 Reading Files

1.18 Files as a Sequence

More resources:

- Exercise

1.19 Python Lists

1.20 Working with Lists

1.21 Strings and Lists

More resources:

- Exercise

1.22 Python Dictionaries

1.23 Dictionaries: Common Applications

1.24 Dictionaries and Loops

More resources:

- Exercise

1.25 The Tuples Collection

1.26 Comparing and Sorting Tuples

More resources:

- Exercise

- 1.27 Regular Expressions
- 1.28 Regular Expressions: Matching and Extracting Data
- 1.29 Regular Expressions: Practical Applications
- 1.30 Networking with Python
- 1.31 Networking Protocol
- 1.32 Networking: Write a Web Browser
- 1.33 Networking: Text Processing
- 1.34 Networking: Using urllib in Python
- 1.35 Networking: Web Scraping with Python

More resources:

- Exercise: socket1
- Exercise: urllib
- Exercise: urllinks

- 1.36 Using Web Services
- 1.37 Web Services: XML
- 1.38 Web Services: XML Schema
- 1.39 Web Services: JSON
- 1.40 Web Services: Service Oriented Approach
- 1.41 Web Services: APIs
- 1.42 Web Services: API Rate Limiting and Security

More resources:

- Exercise: GeoJSON
- Exercise: JSON
- Exercise: Twitter
- Exercise: XML

- 1.43 Python Objects
- 1.44 Objects: A Sample Class
- 1.45 Object Lifecycle
- 1.46 Objects: Inheritance
- 1.47 Relational Databases and SQLite
- 1.48 Make a Relational Database
- 1.49 Relational Database Design
- 1.50 Representing Relationships in a Relational Database
- 1.51 Relational Databases: Relationship Building
- 1.52 Relational Databases: Join Operation
- 1.53 Relational Databases: Many-to-many Relationships

More resources:

- Exercise: Email
- Exercise: Roster
- Exercise: Tracks
- Exercise: Twfriends
- Exercise: Twspider

- 1.54 Visualizing Data with Python
- 1.55 Data Visualization: Page Rank
- 1.56 Data Visualization: Mailing Lists

More resources:

- Exercise: Geodata
- Exercise: Gmane Model
- Exercise: Gmane Spider
- Exercise: Gmane Viz
- Exercise: Page Rank
- Exercise: Page Spider
- Exercise: Page Viz

2 Scientific Computing With Python Projects

2.1 Arithmetic Formatter

Create a function that receives a list of strings that are arithmetic problems and returns the problems arranged vertically and side-by-side.

You can access the full project description and starter code on Repl.it.

After going to that link, fork the project. Once you complete the project based on the instructions in 'README.md', submit your project link below.

We are still developing the interactive instructional part of the Python curriculum. For now, here are some videos on the freeCodeCamp.org YouTube channel that will teach you everything you need to know to complete this project:

Python for Everybody Video Course (14 hours)

Learn Python Video Course (2 hours)

2.2 Time Calculator

Write a function named "add_time" that can add a duration to a start time and return the result.

You can access the full project description and starter code on Repl.it. After going to that link, fork the project. Once you complete the project based on the instructions in 'README.md', submit your project link below.

We are still developing the interactive instructional part of the Python curriculum. For now, here are some videos on the freeCodeCamp.org YouTube channel that will teach you everything you need to know to complete this project:

Python for Everybody Video Course (14 hours)

Learn Python Video Course (2 hours)

2.3 Budget App

Create a "Category" class that can be used to create different budget categories.

You can access the full project description and starter code on Repl.it.

After going to that link, fork the project. Once you complete the project based on the instructions in 'README.md', submit your project link below.

We are still developing the interactive instructional part of the Python curriculum. For now, here are some videos on the freeCodeCamp.org YouTube channel that will teach you everything you need to know to complete this project:

Python for Everybody Video Course (14 hours)

Learn Python Video Course (2 hours)

2.4 Polygon Area Calculator

In this project you will use object oriented programming to create a Rectangle class and a Square class. The Square class should be a subclass of Rectangle and inherit methods and attributes.

You can access the full project description and starter code on Repl.it.

After going to that link, fork the project. Once you complete the project based on the instructions in 'README.md', submit your project link below.

We are still developing the interactive instructional part of the Python curriculum. For now, here are some videos on the freeCodeCamp.org YouTube channel that will teach you everything you need to know to complete this project:

Python for Everybody Video Course (14 hours)

Learn Python Video Course (2 hours)

2.5 Probability Calculator

Write a program to determine the approximate probability of drawing certain balls randomly from a hat. You can access the full project description and starter code on Repl.it. After going to that link, fork the project. Once you complete the project based on the instructions in 'README.md', submit your project link below.

We are still developing the interactive instructional part of the Python curriculum. For now, here are some videos on the freeCodeCamp.org YouTube channel that will teach you everything you need to know to complete this project:

Python for Everybody Video Course (14 hours)

Learn Python Video Course (2 hours)