

APPENDIX

Next

If you've made it to this chapter, congrats! I hope you've learned something!

If you enjoyed this book and got a thrill out of learning about Python programming, I encourage you to keep at it. It's a really great language. If you want to learn more, plenty of wonderful books and resources are available. This appendix gives a few recommendations.

Illustrated Guide to Python 3 by Matt Harrison

This book brings developers and others who are anxious to learn Python up to speed quickly. Not only does it teach the basics of syntax, but it condenses years of experience. You will learn warts, gotchas, best practices, and hints that have been gleaned through the years by the author. You will hit the ground running in the right way.

The Hitchhiker's Guide to Python by Kenneth Reitz and Tanya Schlusser

This book takes the journey person Pythonista to true expertise. More than any other language, Python was created with the philosophy of simplicity and parsimony. Now 25 years old, Python has become the primary or secondary language (after SQL) for many business users. With popularity comes diversity—and possibly dilution.

Think Python by Allen B. Downey

This book is an introduction to computer science using the Python programming language. It covers the basics of computer programming, including variables and values, functions, conditionals and control flow, program development, and debugging. Later chapters cover basic algorithms and data structures.

Python for Data Analysis: Data Wrangling with Pandas, Numpy, and IPython by Wes McKinney

Looking for complete instructions on manipulating, processing, cleaning, and crunching structured data in Python? The second edition of this hands-on guide—updated for Python 3.5 and pandas 1.0—is packed with practical cases studies that show you how to effectively solve a broad set of data analysis problems, using Python libraries such as NumPy, pandas, Matplotlib, and IPython.

Python Data Science Handbook: Tools and Techniques for Developers
by Jake VanderPlas

For many researchers, Python is a first-class tool mainly because of its libraries for storing, manipulating, and gaining insight from data. Several resources exist for individual pieces of this data science stack, but only with this book do you get them all—IPython, NumPy, pandas, Matplotlib, scikit-learn, and other related tools.

Finally, please e-mail me if you have any concerns, questions, or comments about this book. Your feedback is tremendously valuable, and I will do my best to respond to each e-mail. Again, I can be reached via e-mail.

max{dot}humber{at}gmail{dot}com

I look forward to hearing from you!

Index

A

- Alpha Vantage
 - API key, 92
 - TIME_SERIES_DAILY_
ADJUSTED endpoint, 93
- am function, 49
- Amortization, 41
 - evaluate, 49–51
 - functionize, 48
- Anaconda, 1–4

B

- Banks, 40
- Bitcoin, 9
- Budget
 - adding vacation, 65–66
 - cash flow, 53
 - dates, 53
 - flows, 58
 - cash flow objects, 60
 - date_range function, 60
 - .fillna(0) method, 61
 - month start, 59
 - semi-month end, 59
 - fun object, 73
 - functionize, 76

- time horizon, 57
 - pd.date_range function, 58
- timestamp
 - date-time.datetime
object, 55
- totals
 - cumsum(), 61
 - .tail method, 62
- updating, 63–65
- visualization, 63
 - vanilla matplotlib, 62
- YAML, 74

C

- CAD to USD, converting
 - documentation, 28–31
 - encapsulate, 31–33
 - .apply, 34–38
 - show_alternative, 33
 - openexchangerates.org, 26
 - secrets, 27–28
- Calendar object, 75
- Canadian dollars (CAD), 25
 - See also* CAD to USD,
converting
- Computer programming, 112

D, E, F, G, H

Data, 8

Dates

- date formatting rules, 54
- datetime.date objects, 55
- datetime.datetime objects, 55
- get_dates function, 69–70
- Python, 54

DatetimeIndex object, 71–72

Dogecoin, 9

- IRR, 12
- =IRR(), 12–14
- irregular cash flow
 - schedule, 22–24
- mining, 10–11
- pandas
 - read_excel function, 15–17
 - xirr function, 20–22
 - xnpv function, 17–18, 20
- ROI, 11

I

Internal rate of return (IRR), 12

Investment portfolio, 100

- deposit function, 90
- design, 82
 - adding cash, 85
 - .at method, 86
 - DataFrame, 83
 - instantiate_portfolio function, 85
 - reusable function, 84

get_order function, 88–89

prices

- gaps, 87
- print(portfolio) function, 88
- .update method, 87
- rebalance, 101–102
- simulate_process_order function, 91
- stock quotes access
 - Alpha Vantage API, 92
 - get_historical function, 98, 100
 - get_price function, 95, 97–98

J, K, L

Jupyter, 5

M

Month start frequency, 58

N, O

nteract, 5

- blank state, 6
- macOS, 7
- pip install, 8

P, Q

Pandas, 82

- DataFrame, .from_dict code, 85
- Series, 71

Pandas 1.0, 112

Payment, 41

 loop A, 42–45

 loop B, 46–48

Personal investment portfolio, 81

Prophet, 104

 definition, 103

 forecast

 datestamp column, 105

 .make_future_dataframe

 method, 106

 numeric column, 105

 predict method, 106

 purchases, 105

 purchases.csv file, 104

 visualize, 108

 plot convenience method, 107

 plot_components

 method, 108

Python

 floor division operator, 89

 forecasting

 libraries, 112

 programming, 111

R

Recurrence rule (rrule), 67

 .between method, 68

Recurrent library, 67, 69

S

Semi-month-end

 frequency, 58

T

Time Series (Daily) key, 97

Time-series forecasting, 103

Timestamp

 normalizing, 56

 .normalize method, 56

 to_datetime function, 56

U

United States dollar (USD), 25

See also CAD to USD,
 converting

V, W, X

Vacation budget, 77–79

Y, Z

YAML

 Anaconda, 74

 loading, 79

 with block, 80

 totals, 75