front matter

reface

When I started to teach Python at companies around the world, I wasn't surprised by how my students were using the language. They were typically using it the same way I was: for shell scripting in a more expressive language than Bash, writing server-side web applications, developing automated tests, and working with relational databases.

After a while, I found that students were using Python to analyze data—something I hadn't expected. Python was powerful and easy to use, but it was also fairly inefficient. How could people use it for data analysis?

I soon learned what many others already knew: NumPy combined the ease of Python with the efficiency of C. I jumped on the NumPy bandwagon, using it for

analysis and teaching courses in it. But NumPy was still a bit too low-level for my tastes.

I was thus delighted to discover pandas, which gave me the speed and efficiency of NumPy but with a rich API that made many of my daily tasks easier. I have often described pandas as being like a car's automatic transmission, which we can contrast with the low-level manual transmission that NumPy provides. Pandas allowed me to read and write data in a variety of formats, to examine and analyze my data, to clean it, and to visualize it—in short, all the functionality I needed. I was hooked.

In the decade since I first encountered pandas, interest in the library has skyrocketed. It's hard to exaggerate the degree to which pandas is now being used; I've personally taught pandas everywhere from government agencies to startups to hedge funds to Fortune 100 companies.

Pandas approaches problems differently than Python. The syntax is the same, but the

data structures are different, and the way you structure your solutions is also different. Pandas is so vast that it's easy to lose track of all the techniques. And unlike the core Python language, which tries to adhere to the maxim "There should be only one way to do it," there are often many ways to accomplish the same task in pandas. Knowing which of these ways is fastest to execute and easiest to maintain isn't always obvious, even (or especially) if you're an experienced Python developer.

For all these reasons, I'm a big believer in practice. Only by practicing the use of pandas can you remember its most important functionality and know how to apply it. And it's not enough to practice with pretend, synthetic data; if you want to really get good with pandas, you need to use real-world data with all its problems, warts, missing values, and poor construction.

The exercises in this book all come from classes I've taught over the last decade. Many have gone through iterations

and changes along the way as I've seen what problems newcomers to pandas experience and the kinds of problems most likely to trip people up. My goal is to give you an opportunity to practice your pandas skills in a way that sets you up for success when you use pandas at work. Just as every run of a flight simulator makes a pilot more ready to fly an airplane full of passengers, every exercise you do in this book will make you more ready to use pandas to its fullest potential.

cknowledgments

A large number of people have helped me put together *Pandas Workout*.

Although my name appears on the cover, many people at Manning Publications have given me incredible (and patient) support during its creation. Chief among them are associate publisher Mike Stephens, who encouraged me to write a second book, and editor Frances Lefkowitz, who knows how to provide just the right amount of gentle pressure along with useful

editorial suggestions. I received helpful comments from technical reviewer Ninoslav Cerkez as well.

Several dozen people signed up to read, review, and comment on the book while it was being written and edited. Their comments definitely helped me improve and sharpen the text, code, examples, and explanations. I also greatly appreciate the many people who bought *Pandas Workout* in the prerelease (MEAP) form and who commented on Manning's liveBook system.

I am grateful to the team that produces the Pandas Tutor website for providing interactive visualization of pandas queries in the same way the Python Tutor site does for Python programs. The link following each exercise in this book takes you to a pandas Tutor page prefilled with my solution. The nature of pandas, and of Pandas Tutor, means I had to make do with truncated data—but the visualization will still help you better understand the solution.

Thank you to all the reviewers —Alain Couniot, Alex Garrett, Alex Lucas, Alexander Kogler, Amilcar de Abreu Netto, Cage Slagel, Dean Langsam, George Mount, Helen Mary Labao Barrameda, Jeff Neumann, Jeff Smith, Juan Delgado, Kiran Anantha, Mikael Dautrey, Miki Tebeka, Răducu Sergiu Popa, Sadhana Ganapathiraju, Salil Athalye, Satej Kumar Sahu, Sruti Shivakumar, Steven Herrera, and Xiangbo Mao—your suggestions helped make this a better book.

Finally, my family has been incredibly patient, somehow believing me every time I told them I had “just a few more things to edit” as I wrote the book over the past three years. Thanks so much to my wife, Shira, and our three children, Atara, Shikma, and Amotz.

bout this book

Collecting data used to be a challenge. That’s no longer the case, thanks to small, cheap sensors, ubiquitous mobile devices, and the integration of computing into nearly every part of our lives. Now our world is awash in more data

than we know what to do with, tracking everything from the steps we take to the effectiveness of advertising to the temperature on nearly any part of the planet.

We're now faced with a new problem: how can we sort through all this data we've collected? How can we make sense of it and use it to make better decisions?

For decades, the go-to choice has been Microsoft Excel. This makes sense; Excel is convenient, graphical, and installed on nearly every computer in the world. Excel makes it fairly easy to import data, clean it, perform calculations with it, and produce fancy, colorful reports, including charts.

In the last few years, though, Excel has faced a new and surprising challenger: pandas. Pandas started as a convenient wrapper for NumPy, a library that combines the speed and efficiency of C with the friendliness of Python. Pandas added many methods to NumPy's offerings, including high-quality support for text

strings, date/time data, and visualization. Pandas can also read and write data in a wide variety of formats, including from online resources and relational databases.

All this, along with the underlying power of the Python language, the fact that pandas can handle far larger data sets than Excel, and its ability to run “headless” rather than take up an individual analyst’s computer, has increasingly tipped the scales in favor of pandas. I’ve taught Python and pandas at numerous financial institutions that are moving their analysts away from Excel and toward pandas for these reasons, and I’ve worked with many companies in other sectors that are increasingly standardizing on pandas.

Of course, Excel isn’t the only tool or language for data analysis. People are moving to pandas from programming languages like R and Matlab, too—partly for the price, partly for the performance, and partly for the huge ecosystem of open source Python modules available on

the Python Package Index (PyPI).

The problem is that pandas is a *huge* library with thousands of methods and numerous options that you can pass to each of them. And pandas offers numerous ways to accomplish a given task, one of which is often much more performant than the others.

Learning how to work with pandas and how to use it correctly and efficiently frequently means a great deal of trial and error. A shortcut to mastery is to practice on problems specifically meant to help you better understand specific pandas features, much as particular exercises are meant to tone specific muscles.

That's where this book comes in. Across 50 main exercises (and 150 more "Beyond the exercise" challenges, as well as two larger projects), *Pandas Workout* will make you a more fluent, confident user of pandas. Each exercise asks you to load real-world data into pandas and then answer various questions about that data. As you work through the

book, you'll learn about the most important parts of pandas—and even more importantly, you'll learn how and when it's appropriate to use them.

Pandas Workout isn't designed to teach you pandas, although I hope you'll learn quite a bit along the way. Rather, this book is meant to help you improve your understanding of pandas, how it works, and how to use it to answer questions based on data.

Please don't just read through the book. It's also a mistake to read an exercise, say to yourself that you know how to solve it, and then move on. Each exercise includes several questions, and many of them are trickier to answer than you may think. Moreover, reading my solutions without having worked on the exercises yourself isn't nearly as effective for internalizing how pandas works. So, please take the time to do the exercises, working through them gradually.

You should especially avoid feeding my questions into

ChatGPT and just reviewing the answers it gives. Not only are those answers often wrong, but real learning comes from struggling a bit, getting things wrong, and then learning from your mistakes.

Who should read this book

If you've taken a pandas course but are still searching on Stack Overflow or Google for how to solve problems with pandas, this book is for you. It's not a tutorial but is meant to solidify your understanding of pandas via repeated practice.

Many pandas courses don't emphasize the need for core Python knowledge before learning pandas. I firmly believe you should get a good grounding in Python if you'll be using pandas, and this book reflects that perspective. However, you don't need to know *that* much; I assume you're comfortable with core data types, loops, functions, list comprehensions, and installing modules with `pip`. In a few places (not too many), you can also benefit from knowing about `lambda`.

How this book is organized: A road map

This book has 13 chapters, each focusing on a different aspect of pandas. Exercises in each chapter use techniques from previous chapters and sometimes from later ones. For example, we use string techniques (chapter 9) and `datetime` values (chapter 10) in earlier chapters. Think of the titles as general guidelines, rather than strict rules, for what you'll practice and learn in each chapter.

The chapters cover these topics:

1. *Series*—Understanding what a series is and how we can retrieve selected values from a series.
2. *Data frames*—Constructing data frames and retrieving selected values from a data frame.
3. *Import and export*—Reading and writing files in different formats, including CSV and JSON.
4. *Indexes*—Setting and retrieving indexes and multi-indexes.

5. *Cleaning*—Turning messy, real-world data into a form we can use more easily: for example, identifying duplicates, handling missing values, and removing unnecessary and incorrect data.
6. *Grouping, joining, and sorting*—The core of much pandas functionality: grouping data, joining multiple data frames, and sorting by both indexes and values. These topics are so important that two chapters address them.
7. *Advanced grouping, joining, and sorting*—Deeper examination of the techniques introduced in chapter 6.
8. *Project*—Completing a large project based on the Python developer survey.
9. *Strings*—Working with text data from within pandas.
10. *Dates*—Working with date and time data from within pandas.
11. *Visualization*—Plotting both via the pandas API and using the Seaborn module.
12. *Performance*—Optimizing the speed and memory usage of our data.

13. *Final project*—Completing a large project examining American colleges and universities.

Exercises form the main part of each chapter. Each exercise has five components:

- *Exercise*—A problem statement for you to tackle.
- *Working it out*—A detailed discussion of the problem and how to solve it.
- *Solution*—The solution code and (in most cases) a link to the code on the Pandas Tutor site so you can execute it. Solution code, along with test code for each solution, is also available on the Manning website at **www.manning.com/books/pandas-workout** and GitHub at **https://github.com/reuven/pandas-workout**.

- *Beyond the exercise*—Three additional, related exercises. These questions are neither answered nor discussed in the book, but the code is downloadable along with all the other solution code from the book. You can also discuss these additional exercises and compare solutions with other *Pandas Workout* readers in the book's online forum on Manning's liveBook platform.

About the code

This book contains a great deal of pandas code. Unlike most books, the code reflects what you are supposed to write rather than what you're supposed to read. If experience is any guide, some readers (maybe you!) will have better, more elegant, or more correct solutions than mine. If this is the case, don't hesitate to contact me.

Solution code for all exercises, including the "Beyond the exercise" questions, is available in these places outside of the book:

- The *Pandas Workout* website (**www.manning.com/books/pandas-workout**) and GitHub repo (**https://github.com/reuven/pandas-workout**) have all the code solutions organized by chapter and then by exercise number so you can download the code and run it on your own computer.
- Pandas Tutor (**https://PandasTutor.com**), an amazing online resource for teaching and learning pandas, allows you to enter nearly any pandas code and see how it works, with visual cues demonstrating transformations. Most of the solutions in this book have a link pointing to the code in the Pandas Tutor so you can run it without typing it into the site. Note that those links generally use small samples of the data.

This book contains many examples of source code, both in numbered listings and in line with normal text. In both cases, the source code is formatted in a fixed-width

font `like this` to separate it from ordinary text.

In many cases, the original source code has been reformatted; we've added line breaks and reworked indentation to accommodate the available page space in the book. In rare cases, even this was not enough, and listings include line-continuation markers (➥). Additionally, comments in the source code have often been removed from the listings when the code is described in the text. Code annotations accompany many of the listings, highlighting important concepts.

I hope that the combination of the solution code (in print), explanations, Pandas Tutor links, and downloadable code will help you fully understand each solution and apply its lessons to your own code.

Software/hardware requirements

First and foremost, this book requires that you have both Python and pandas. You can download and install Python most easily from

https://python.org. I suggest installing the latest version available. There are also other ways to install Python, including the Windows Store or Homebrew for Mac. This book should work with any version of Python from 3.9 and up; I used 3.12 in the final checks of the code.

You also need to install pandas. I used pandas 2.1.4 by the time the book was done, but most or all of the code should work fine with any 2.1.x version. You can download and install it using `pip install pandas` on the command line.

You aren't required to install an editor or IDE (integrated development environment) for Python, but it will certainly come in handy. Two of the most popular IDEs are PyCharm (from JetBrains) and Visual Studio Code (from Microsoft). I'm a big fan of the Jupyter Notebook, which you can install with `pip install jupyter`.

liveBook discussion forum

Purchase of *Pandas Workout* includes free access to liveBook, Manning's online reading platform. Using liveBook's exclusive discussion features, you can attach comments to the book globally or to specific sections or paragraphs. It's a snap to make notes for yourself, ask and answer technical questions, and receive help from the author and other users. To access the forum, go to **https://livebook.manning.com/book/pandas-workout/discussion**. You can also learn more about Manning's forums and the rules of conduct at **https://livebook.manning.com/discussion**.

Manning's commitment to our readers is to provide a venue where a meaningful dialogue between individual readers and between readers and the author can take place. It is not a commitment to any specific amount of participation on the part of the author, whose contribution to the forum remains voluntary (and unpaid). We suggest you try asking the author some challenging questions lest his

interest stray! The forum and the archives of previous discussions will be accessible from the publisher's website as long as the book is in print.

bout the author

Reuven M. Lerner is a full-time Python and pandas trainer, teaching both companies and individuals in person and online. Reuven also publishes "Better Developers," a weekly newsletter about Python, and "Bamboo Weekly," with pandas challenges based on current events. Reuven holds a bachelor's degree in computer science from MIT and a PhD in learning sciences from Northwestern. He also wrote *Python Workout*, published by Manning in 2020.

bout the cover illustration

The figure on the cover of *Pandas Workout* is “Femme Tongouse,” or “Woman of Tunguska, Northern Siberia,” taken from a collection by Jacques Grasset de Saint-Sauveur, published in 1788. Each illustration is finely drawn and colored by hand.

In those days, it was easy to identify where people lived and what their trade or station in life was just by their dress. Manning celebrates the inventiveness and initiative of the computer business with book covers based on the rich diversity of regional culture centuries ago, brought back to life by pictures from collections such as this one.