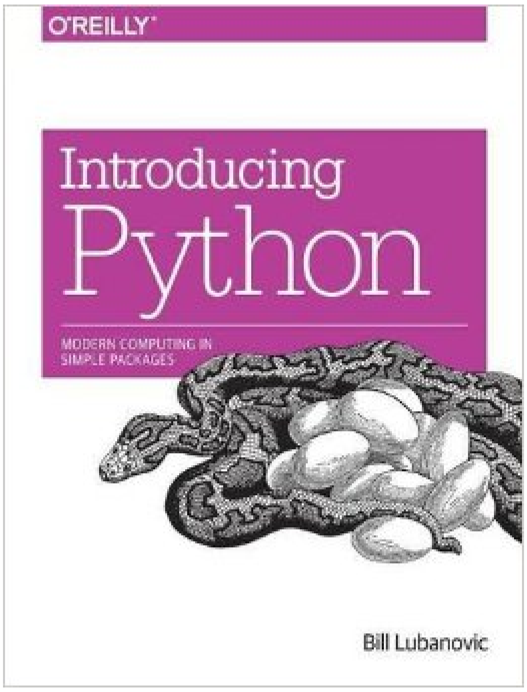
Introducing Python: Modern Computing In Simple Packages



Synopsis

Easy to understand and fun to read, Introducing Python is ideal for beginning programmers as well as those new to the language. Author Bill Lubanovic takes you from the basics to more involved and varied topics, mixing tutorials with cookbook-style code recipes to explain concepts in Python 3. End-of-chapter exercises help you practice what youâ ™ve learned.Youâ ™ll gain a strong foundation in the language, including best practices for testing, debugging, code reuse, and other development tips. This book also shows you how to use Python for applications in business, science, and the arts, using various Python tools and open source packages.Learn simple data types, and basic math and text operationsUse data-wrangling techniques with Pythonâ ™s built-in data structuresExplore Python code structure, including the use of functionsWrite large programs in Python, with modules and packagesDive into objects, classes, and other object-oriented featuresExamine storage from flat files to relational databases and NoSQLUse Python to build web clients, servers, APIs, and servicesManage system tasks such as programs, processes, and threadsUnderstand the basics of concurrency and network programming

Book Information

Paperback: 484 pages
Publisher: O’Reilly Media; 1 edition (December 4, 2014)
Language: English
ISBN-10: 1449359361
ISBN-13: 978-1449359362
Product Dimensions: 7 x 1 x 9.2 inches
Shipping Weight: 1.7 pounds (View shipping rates and policies)
Average Customer Review: 4.5 out of 5 starsÂ Â See all reviewsÂ (33 customer reviews)
Best Sellers Rank: #25,455 in Books (See Top 100 in Books) #37 inÂ Books > Computers & Technology > Programming > Languages & Tools > Python #67 inÂ Books > Computers & Technology > Programming > Web Programming #104 inÂ Books > Textbooks > Computer Science > Programming Languages

Customer Reviews

This is an insightful and fast-paced intro to Python, written by an experienced master coder. It’s written for coders (unlike the plethora of basic Python books on the market). The very first chapter presents a non-trivial 9-line Python program that pulls and prints top-rated videos from the Youtube site (the code works). The book covers many of the language features as well as areas such as web programming, databases, and concurrency. The book is a pleasure to browse through: an intelligent and informed writer writing for an intelligent readership. After reading this book one will be ready for Wesley Chun's "Core Python Applications Programming."

I've just started reading, and already I've encountered a pretty big error regarding substring indexing on page 33. Be sure to check the errata as you progress through this book.http://www.oreilly.com/catalog/errata.csp?isbn=0636920028659

I'd agree with the comments made by others. It's really an excellent introduction to the language, done in a very modern way, which is to recognize that most coding is now done by gluing together modules. Because this book takes that approach, and uses the modules as a foil for teaching you how to code in a Pythonic way, you really learn two things at once.If you have any coding experience, and want to get up and running quickly, this is a great book, and even a good reference for later projects as they arise.And yes, "Learning Python" is really awful (both on its own terms, and in comparison to this). This is a MUCH better place to start. (What happened there anyway O'Reilly, I mean Learning Perl is THE book.BACKGROUND: I'm an experienced coder, having learned C and C++ years ago, giving them up when I learned Perl, and figured I'd never parse a file byte-by-byte ever again. I learned Ruby more recently, but got into some heavy math/statistics/image analysis, and saw that Python had the best supported science/math libraries by far. So I had to learn it. In spite of the fact that I think regular expressions are far harder in Python then they need to be, I see the advantages that other list. But most of all, most of all, it's about the available modules. CPAN is what made Perl unbeatable back in the day, and Python is the modern-day heir to that legacy. This book does a good job recognizing that and making you fluent in the language and it's highest profile modules at the same time. Highly recommended. (And no, you can't borrow my copy).

Review of "Introducing Python" By Bill LubanovicReviewed by Alfredzo Nash, Fairfield County Area Datto Linux User Group."Introducing Python" by Bill Lubanovic is an edible recipe for learning Python. I had limited exposure to Python (I completed 50% of the Codecademy training) prior to reading "Introducing Python". Lubanovic is ab iron chef that has written this delectable book.Beginning with Chapters 1-5 (1. A Taste of Python, 2. Python Ingredients, 3. PyFilling, 4. PyCrust, and 5. PyBoxes), the reader starts to build their mental palate using a small cookbook style dialog for making an actual pie! This was very informative and one of the best analogies within the book. Lubanovic skillfully separates the importance of data structure such as list, dictionaries, tuples, and set from code structure (commenting, if elif else, utf-8 encoding, pep, etc.) which allowed me to become familiar with the essential building blocks of the Python language. "Introducing Python" is a very definitive guide to what makes Python a valuable and powerful language. Since Python 2.7's end-of-life is scheduled in 2020, Lubanovic encouraged Pythonistas to begin writing Python code using Python 3 instead of 2.7. Being relatively new to the language, I couldn't resist the urge to discover more about the differences between the two versions. At the time of this writing, there are still subtle differences between Python 2.7 and Python 3. The changes include how to call the print() function, and the handling of Unicode characters. Most of the differences between version 2.7 and 3.0 can be found at https://wiki.python.org/moin/Python2orPython3.Unfortunately, Chapters 6-9 (6. Oh Oh: Objects and Classes, 7. Mangle Data Like a Pro, 8. Data Has to Go Somewhere, 9. The Web Untangled) proved difficult for me to comprehend due to my inexperience. Lubanovic attempts to ease the reader into this section skillfully in Chapter 6, with analogies such as "In Chapter 1, I compare an object to a plastic box. A class is like the mold that makes that box." I couldn't agree with this analogy as I had to learn the complex object-oriented terminology, like polymorphism, instantiation and inheritance, discussed throughout this chapter.Chapters 7-8 puts "Introducing Python" into high gear. I did not take the term "Pro" lightly within this section. Essentially, I now understand why data scientist, forensic analysts, cloud engineers, and automation engineers consider Python for their projects. Data can take many forms that include text strings, ASCII or binary. These types of data need to be written, read, encoded, decoded and stored properly. Lubanovic clearly conveyed the message that data should be handled with precision. UTF-8 encoding, byte arrays, and matching regular expression all took some time for me to understand. There are tons of options that include storing data between RAM, CSVs, JSON, or SQL and NoSQL Databases. Both mangling and storing data requires a certain level of mastery that has convinced me to keep Introducing Python as a reminder to sharpen my skills in this area of expertise.Chapters 9-12 (9.The Web, Untangled, 10. Systems, 11. Concurrency and Networks,12. Be a Pythonista), are my favorite chapters in Introducing Python. Chapter 9 explains how Python's standard Web libraries handle the various components of the Web. It describes mainly the http and urlib packages, but also makes notable references to web frameworks such as Bottle and Flask. Chapter 10 dived into how Python handles files, directories, processes and time using the os (Operating System) module. The os module contains the functions copy(),chown(), remove(), mkdir(),rmdir() and many more that operate the same way as their Linux/UNIX counterparts. Chapter 11 reinforces Chapter 10's section on process management via the os module and using multiprocessing. Diving deeper into the concurrency standard library (http://bit.ly/concur-lib) within

Python, Lubanovic discusses the differences between queues, process, and threads, as well as the tools to manage them on a single machine or multiple machines. Noticabley, Redis, gevent, asyncio, and twisted are mentioned as additional tools that handle concurrency within Python as well. In the Networks section, Lubanovic explores the plumbing provided by the numerous Internet Services supported within Python. TCP/IP, Sockets, Web APIs (fabric), RPCs and Message Brokers (RabbitMQ /ZeroMQ) are all covered in great detail. This chapter is an invaluable resource for anyone interested in gaining a more in-depth knowledge of how use Python to manage concurrency across distributed systems and networks. Chapter 12 sets the example for ALL Python users, by encouraging them to "Be A Pythonista". Noticeably, Being a Pythonista is no small task. The requirements consist of: Finding Code, Installing Packages, Documenting Your Code, Testing Your code, Optimizing your code, and Managing your code via Source ControlÃƒÂ¢Ã¢Â Â¬Ã Â¦ Whew... Humbly, I accept this challenge and hope to become a proficient Pythonista and encourage everyone interested in Python to do the same.In closing, Introducing Python was remarkable read for me. I recommend studying this book and holding onto for future reference in real-world scenarios.

I agree that this book is excellent. Don’t pick this up as your first Python book unless you are a seasoned coder of another language. It simply moves too fast. As an example, page 81 covers ’Comprehensions’. The author states, ’using a comprehension is sometimes taken as a sign that you know Python at more than a beginner’s level’.

This book is good for experienced programmers to study Python and has plenty of contents tolearn. Packages of various fields are explained, with those of similar function listed. So this bookcan be used for reference. Reader will be exposed to the various computing fields advancing currently.In my own opinion, it may be better to discuss NoSQL database, Queue usage and NetWork with ZeroMQpackage more in real practice scenarios. And the meaningful words and little jokes interspersed in thebook make it more attractive than purely describing a programming language.

Marvelous for everyone, programmers and (I presume) non-programmers alike. I appreciate a smooth, concise introduction to a computer language just as much as anyone; just because I can suffer through learning one the hard way doesn’t mean I want to do that. This was a fun way to quickly get started. For some reason, the online tutorials are just not my thing. Buy this, spend a few hours with it in your favorite reading spot, and then go to the computer and have fun.

Introducing Python: Modern Computing in Simple Packages Python: Python Programming Course: Learn the Crash Course to Learning the Basics of Python (Python Programming, Python Programming Course, Python Beginners Course) Unsupervised Deep Learning in Python: Master Data Science and Machine Learning with Modern Neural Networks written in Python and Theano (Machine Learning in Python) Convolutional Neural Networks in Python: Master Data Science and Machine Learning with Modern Deep Learning in Python, Theano, and TensorFlow (Machine Learning in Python) Deep Learning in Python: Master Data Science and Machine Learning with Modern Neural Networks written in Python, Theano, and TensorFlow (Machine Learning in Python) Beginning Python Programming: Learn Python Programming in 7 Days: Treading on Python, Book 1 Python: Python Programming For Beginners - The Comprehensive Guide To Python Programming: Computer Programming, Computer Language, Computer Science Learn Python in One Day and Learn It Well: Python for Beginners with Hands-on Project. The only book you need to start coding in Python immediately Maya Python for Games and Film: A Complete Reference for Maya Python and the Maya Python API Python: Python Programming For Beginners - The Comprehensive Guide To Python Programming: Computer Programming, Computer Language, Computer Science (Machine Language) Deep Learning: Recurrent Neural Networks in Python: LSTM, GRU, and more RNN machine learning architectures in Python and Theano (Machine Learning in Python) Deep Learning in Python Prerequisites: Master Data Science and Machine Learning with Linear Regression and Logistic Regression in Python (Machine Learning in Python) Introducing Gershwin (IC) (Introducing Composers) Introducing Literary Criticism: A Graphic Guide (Introducing Graphic Guides) Introducing Oceanography (Introducing Earth and Environmental Sciences) Introducing Lacan: A Graphic Guide (Introducing...) Introducing Logic: A Graphic Guide (Introducing...) Introducing Postmodernism: A Graphic Guide (Introducing...) Introducing Islam: A Graphic Guide (Introducing...) Introducing Slavoj Zizek: A Graphic Guide (Introducing...)