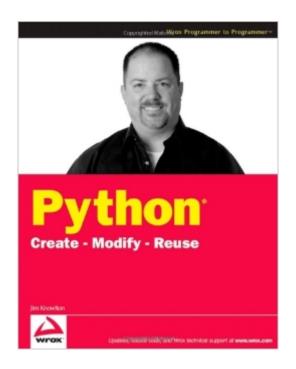# Python: Create – Modify – Reuse

Synopsis

Python: Create-Modify-Reuse is designed for all levels of Python developers interested in a practical, hands-on way of learning Python development. This book is designed to show you how to use Python (in combination with the raw processing power of your computer) to accomplish real-world tasks in a more efficient way. Don't look for an exhaustive description of the Python languageâ "â "you won't find it. The book's main purpose is not to thoroughly cover the Python language, but rather to show how you can use Python to create robust, real-world applications. This book is for developers with some experience with Python who want to explore how to develop full-blown applications. It is also for developers with experience in other languages who want to learn Python by building robust applications. It is well-suited for developers who like to â œelearn by doing,â • rather than exploring a language feature by feature. To get the most out of the book, you should understand basic programming principles. This book starts with a basic overview of the Python language, designed for those familiar with other languages but new to Python. It is followed by several chapters, each of which describes a complete project that can be used as-is or modified and extended to suit your particular purposes. You'll find applications that access databases, take advantage of web technologies, and facilitate network communications, to name a few. In addition, and more important than the technologies you will be introduced to, you will learn how to use Python to solve real challenges. Following these chapters are two chapters that cover accessing operating system resources and debugging and testing, respectively. This book is framed around the code itself. This is because developers are typically looking for how to do something; and, as with many activities, you learn how to do something by watching how others do and trying it yourself. If you want to know how a for loop works, you'll find for loops in my code, but that's not the thrust of the book. Instead, this book shows you how to do things: how to build a content management system, how to build a test management system, how to set up a system for tracking customer follow-up, and so on. Along the way, you'll learn how to communicate with a SQL database, how to act as a web server or communicate with one, how to access operating system services, and more.

Book Information

Product Dimensions: 7.4 x 0.6 x 9.3 inches

Shipping Weight: 12.8 ounces

Average Customer Review: 3.2 out of 5 starsÂ Â See all reviewsÂ (6 customer reviews)

Best Sellers Rank: #3,010,842 in Books (See Top 100 in Books)   #24 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Software Reuse   #1324 inÂ Books > Computers & Technology > Programming > Languages & Tools > Python   #2949 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Object-Oriented Design

Customer Reviews

I really wanted to like this book. As a professional C# developer, I have been learning Python the last few weeks. I got this book in hopes that it would provide some interesting projects.I could not bring myself to like this book. It was a painful read because this book demonstrates poor and obsolete programming practices. I feel like this book would probably set a beginner back instead of helping them move forward. I would recommend almost any other Python book over this one.The book does have seven fairly interesting projects, covering a wide breadth of the Python standard library (xml parsing, database access, file input/output, etc) and some web-based projects. That is why I wanted to like it. It covers interesting topics. However, this is not enough to make up for the sloppy editing, obsolete user interfaces, and poor programming practices. I will detail some of the problems below.Sloppy editing - For example, look at the code sample at the top of page 31. This wouldn't even "compile" in Python's editor IDLE, because of the incorrect indenting.....elif choice == "4":........snapshothelper.showHelp().........else:........if choice != "5":............snapshothelper.invalidChoice()The "else:" is intended wrong and then on page 32 the same code is excerpted except it has been copied incorrectly:....elif choice == "4":........snapshothelper.showHelp()....else:.........snapshothelper.invalidChoice()You can see that the author corrected a mistake in the main block of code but then did not correct it in the excerpt.This is just the tip of the iceberg for code blocks that will cause errors upon running them inside Python.On to the next problem - The user interface for the command-line projects looks like something out of MS-DOS 3.1. In other words, it is extremely outdated, cumbersome and it's not even programmed effectively. The error messages are obtuse and don't even cover all the cases. Frankly, I expect a lot more from an author with an extensive background in testing. For example, in page 70, the "elif" block never checks for choices that are not 1 through 5. If you have an invalid choice it will cause a SQL error and print the error message "THERE WAS A PROBLEM

MODIFYING THE RECORD" even though the error was caused by an invalid choice.My final complaint is that the author does not follow Pythonic practices or even best practices from other languages. For example, the Try-Except blocks (page 79 for example) include large swaths of code instead of just the small section that will actually throw the exception. Also, the Except clause does not catch particular error classes but instead every error. These violate basic best practices for any language. Another example of not being very Pythonic is the lack of list comprehensions like on page 37 (also note that this sample includes inconsistent indentation and would likely cause the Python interpreter to fail):....for item in filelist:...........if item.find(extension) != -1:...............snaplist.append(item)This could easily be re-written in the Pythonic way:....snaplist = [for item in filelist if item.find(extension) = -1]Unfortunately, I could go on and on talking about things I don't like about this book. The bottom line is that I don't recommend it.

I was very disappointed by this book. I have come to expect more from WROX than this. If I was given this book without a cover, I would never have guessed the title. The book is organized more like a cookbook of projects that each illustrate a particular point or technique than a book titled "Create - Modify - Reuse".The source code in the book is broken into fragments and isn't usable by itself. You will need to download the accompanying source from the publisher to run the programs. Unfortunately too much of the downloaded source is broken and has to be fixed before it will run. The style and organization of the code reminds me of a first year student's programming assignment.I'm just glad I only borrowed this book from the library and it had to go back anyway.

This is not a bad book, it just isn't a book for beginners. There is not a whole lot in the way of troubleshooting and if something that the book has you do does not work you more or less have to hack the code together yourself to get things off the ground. The community site that partners with this book and others in the series is a wasteland really. It doesn't seem to be active with many of the forum section completely barren and the forums sections with discussions have been inactive for several months if not years. In other words don't expect a whole lot of support from anywhere regarding what you read here. All in all though the book does have some interesting coding and can really help broaden intermediate coders perspectives on how to code in Python.

Well, not that bad 1 star.I've tried to find a book that teach me python about, command line interactive, file read/write, CSV & MySQL full application (not a GUI but in the console interactive).All of the examples are using project as a guideline. The code is easy to follow and the

author explain well (IMO, a bit too detail for the some same topic that no needs to explain)So, in the nature of Python, we have to download additional libraries mostly. So, for most the code examples are work for me.I'm happy about this book very much. SInce I'm looking for a book that can teach me Python in a few days and can read + understand how to use python in each project by reading the book mostly. So, I can now move to the more detail books like Core Python or Python in Nut Shell, etc.What I would like the book to have is about using web framework like Django, TurboGears or Pylon. The book use Plone. No question about how good Plone is but the little chapter for it is too superficial for Plone and it can be read from many websites. So, I think it'd be a popular and more lightweight framework like Django, Pylon, TurboGears.If anyone have a few days even far from your computer. You can learn Python easily by the book and for most programmer, it's no need to type (but recommend) since the book describe the code in detail.Recommend for people who want to know python quickly. Then you'll jump to next Python books like me :-)

The approach that the book takes really caters to my learning style i.e. learning by doing. Normally the programs you write in IT books are useless but to my delight, most programs in this book (8 in total) have potential for immediate usage at home and at work, which makes it that much more rewarding to go through them.This is a great way to quickly learn programming for those of you that already know some of the basics. I can't wait to see more books out of the "Create - Modify - Reuse" series.The author also makes it really clear that the first examples are not optimally designed in order to keep things as clear and simple as possible - OOA&D snobs should look elsewhere. Besides, starting a little loose on encapsulation just affords more repetition to reinforce the key Python concepts.

Nice find

[Download to continue reading...](#)

Python: Python Programming Course: Learn the Crash Course to Learning the Basics of Python (Python Programming, Python Programming Course, Python Beginners Course) Python: Create - Modify - Reuse Adobe AIR: Create - Modify - Reuse 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) Unsupervised Deep Learning in Python: Master Data Science and Machine Learning with Modern Neural Networks written 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) 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) Software Reuse Techniques: Adding Reuse to the System Development Process Reuse of Off-the-Shelf Components: 9th International Conference on Software Reuse, ICSR 2006, Torino, Italy, June 12-15, 2006, Proceedings (Lecture Notes in Computer Science) Software Reuse for Dynamic Systems in the Cloud and Beyond: 14th International Conference on Software Reuse, ICSR 2015, Miami, FL, USA, January 4-6, ... (Lecture Notes in Computer Science) Safe and Secure Software Reuse: 13th International Conference on Software Reuse, ICSR 2013,Pisa, Italy, June 18-20, 2013, Proceedings (Lecture Notes in Computer Science) IntAR, Interventions Adaptive Reuse, Volume 03; Adaptive Reuse in Emerging Economies The Fender Stratocaster Handbook, 2nd Edition: How To Buy, Maintain, Set Up, Troubleshoot, and Modify Your Strat Frugal Woodturner, The: Make and Modify All the Tools and Equipment You Need