

## Doing Math With Python

**Learn to Code by Solving Problems is a practical introduction to programming using Python. It uses coding-competition challenges to teach you the mechanics of coding and how to think like a savvy programmer. Computers are capable of solving almost any problem when given the right instructions. That's where programming comes in. This beginner's book will have you writing Python programs right away. You'll solve interesting problems drawn from real coding competitions and build your programming skills as you go. Every chapter presents problems from coding challenge websites, where online judges test your solutions and provide targeted feedback. As you practice using core Python features, functions, and techniques, you'll develop a clear understanding of data structures, algorithms, and other programming basics. Bonus exercises invite you to explore new concepts on your own, and multiple-choice questions encourage you to think about how each piece of code works. You'll learn how to:**

- Run Python code, work with strings, and use variables
- Write programs that make decisions
- Make code more efficient with while and for loops
- Use Python sets, lists, and dictionaries to organize, sort, and search data
- Design programs using functions and top-down design
- Create complete-search algorithms and use Big O notation to design more efficient code

By the end of the book, you'll not only be proficient in Python, but you'll also understand how to think through problems and tackle them with code. Programming languages come and go, but this book gives you the lasting foundation you need to start thinking like a programmer.

Discover easy-to-follow solutions and techniques to help you to implement applied mathematical concepts such as probability, calculus, and equations using Python's numeric and scientific libraries

**Key Features**

- Compute complex mathematical problems using programming logic with the help of step-by-step recipes
- Learn how to utilize Python's libraries for computation, mathematical modeling, and statistics
- Discover simple yet effective techniques for solving mathematical equations and apply them in real-world statistics

**Book Description**

Python, one of the world's most popular programming languages, has a number of powerful packages to help you tackle complex mathematical problems in a simple and efficient way. These core capabilities help programmers pave the way for building exciting applications in various domains, such as machine learning and data science, using knowledge in the computational mathematics domain. The book teaches you how to solve problems faced in a wide variety of mathematical fields, including calculus, probability, statistics and data science, graph theory, optimization, and geometry. You'll start by developing core skills and learning about packages covered in Python's scientific stack, including NumPy, SciPy, and Matplotlib. As you advance, you'll get to grips with more advanced topics of calculus, probability, and networks (graph theory). After you gain a solid understanding of

**these topics, you'll discover Python's applications in data science and statistics, forecasting, geometry, and optimization. The final chapters will take you through a collection of miscellaneous problems, including working with specific data formats and accelerating code. By the end of this book, you'll have an arsenal of practical coding solutions that can be used and modified to solve a wide range of practical problems in computational mathematics and data science. What you will learn**  
**Get familiar with basic packages, tools, and libraries in Python for solving mathematical problems**  
**Explore various techniques that will help you to solve computational mathematical problems**  
**Understand the core concepts of applied mathematics and how you can apply them in computer science**  
**Discover how to choose the most suitable package, tool, or technique to solve a certain problem**  
**Implement basic mathematical plotting, change plot styles, and add labels to the plots using Matplotlib**  
**Get to grips with probability theory with the Bayesian inference and Markov Chain Monte Carlo (MCMC) methods**  
**Who this book is for** This book is for professional programmers and students looking to solve mathematical problems computationally using Python. Advanced mathematics knowledge is not a requirement, but a basic knowledge of mathematics will help you to get the most out of this book. The book assumes familiarity with Python concepts of data structures.

**Doing Math with Python shows you how to use Python to delve into high school-level math topics like statistics, geometry, probability, and calculus. You'll start with simple projects, like a factoring program and a quadratic-equation solver, and then create more complex projects once you've gotten the hang of things. Along the way, you'll discover new ways to explore math and gain valuable programming skills that you'll use throughout your study of math and computer science**

**Demonstrates the programming language's strength as a Web development tool, covering syntax, data types, built-ins, the Python standard module library, and real world examples.**

**The Statistics and Calculus with Python Workshop**

**Computational Mathematics with SageMath**

**Math Coding**

**An Introduction to Building and Breaking Ciphers**

**Introduction to Scientific Programming with Python**

**A comprehensive introduction to mathematics in Python for artificial intelligence applications**

**A Gentle Introduction to Numerical Simulations with Python**

Hemos elaborado 120 programas de Python y mas de 110 ilustraciones en una obra que sera util tanto a estudiantes de ciencias de los primeros cursos universitarios, como a estudiantes y profesores de bachillerato.Puedes descargar gratuitamente un extracto de 78 paginas del libro en pdf desde nuestro sitio web

www.pysamples.com. Los compradores del libro pueden descargar gratuitamente el código de los programas. Todos los programas y gráficas se han elaborado utilizando software libre, que se puede obtener gratuita y legalmente en internet, por lo que no es necesario adquirir ningún software. Se puede programar gratuitamente en Python tanto si eres usuario de Windows como si prefieres Mac o Linux. El libro puede ser utilizado como libro de matemáticas con el que también se puede aprender a programar, o bien como un libro de programación en Python orientada a las ciencias.

With examples and activities that help you achieve real results, applying calculus and statistical methods relevant to advanced data science has never been so easy

**Key Features** Discover how most programmers use the main Python libraries when performing statistics with Python Use descriptive statistics and visualizations to answer business and scientific questions Solve complicated calculus problems, such as arc length and solids of revolution using derivatives and integrals

**Book Description** Are you looking to start developing artificial intelligence applications? Do you need a refresher on key mathematical concepts? Full of engaging practical exercises, *The Statistics and Calculus with Python Workshop* will show you how to apply your understanding of advanced mathematics in the context of Python. The book begins by giving you a high-level overview of the libraries you'll use while performing statistics with Python. As you progress, you'll perform various mathematical tasks using the Python programming language, such as solving algebraic functions with Python starting with basic functions, and then working through transformations and solving equations. Later chapters in the book will cover statistics and calculus concepts and how to use them to solve problems and gain useful insights. Finally, you'll study differential equations with an emphasis on numerical methods and learn about algorithms that directly calculate values of functions. By the end of this book, you'll have learned how to apply essential statistics and calculus concepts to develop robust Python applications that solve business challenges. What you will learn

- Get to grips with the fundamental mathematical functions in Python
- Perform calculations on tabular datasets using pandas
- Understand the differences between polynomials, rational functions, exponential functions, and trigonometric functions
- Use algebra techniques for solving systems of equations
- Solve real-world problems with probability
- Solve optimization problems with derivatives and integrals

**Who this book is for** If you are a Python programmer who wants to develop intelligent solutions that solve challenging business problems, then this book is for you. To better grasp the concepts explained in this book, you must have a thorough understanding of advanced mathematical concepts, such as Markov chains, Euler's formula, and Runge-Kutta methods as the book only explains how these techniques and concepts can be implemented in Python.

Neighborhood Cleanup guides students as they conceive and set up a neighborhood cleanup with their friends and community. The considerate text includes easy-to-follow lists and will hold the readers' interest, allowing for successful mastery and comprehension. Written with a high interest level to appeal to a more mature audience, these books maintain a lower level of complexity with clear visuals to help struggling readers along. A table of contents, glossary with simplified pronunciations, and index all enhance achievement and comprehension.

This book is designed for middle school students and new programming language learners. Computer science has continuously escalated in popularity over the last decade, as students are increasingly showing interest in coding at a young age. In this book you will find a total of 150 math questions, ranging in difficulty from beginner to advanced, with accompanying Python programming language solutions. Python is one of the most popular coding languages and is comparatively easy to learn. With this book, students will be able to increase their proficiency in coding and math computing. This book can be used as a reference for math and computer science teachers for interdisciplinary purposes and will help students improve their skills and critical thinking.

Programming for Computations - Python

Practical recipes for solving computational math problems using Python programming and its libraries

A Programmer's Introduction to Mathematics

Doing Math with Python

Impractical Python Projects

Practical Programming for Total Beginners

The Mystery Queen

***Doing Math with Python teaches you how to use Python as a tool to delve into math concepts.***

***A Programmer's Introduction to Mathematics uses your familiarity with ideas from programming and software to teach mathematics. You'll learn about the central objects and theorems of mathematics, including graphs, calculus, linear algebra, eigenvalues, optimization, and more. You'll also be immersed in the often unspoken cultural attitudes of mathematics, learning both how to read and write proofs while understanding why mathematics is the way it is. Between each technical chapter is an essay describing a different aspect of mathematical culture, and discussions of the insights and meta-insights that constitute mathematical intuition. As you learn, we'll use new mathematical ideas to create wondrous programs, from cryptographic schemes to neural networks to hyperbolic tessellations. Each chapter also contains a set of exercises that have you actively explore mathematical topics on your own. In short, this book will teach you to engage with mathematics. A Programmer's Introduction to Mathematics is written by Jeremy Kun, who has been writing about math and programming for 10 years on his blog "Math Intersect Programming." As of 2020, he works in datacenter optimization at Google. The second edition includes revisions to most chapters,***

*some reorganized content and rewritten proofs, and the addition of three appendices.*

*Learn math by getting creative with code! Use the Python programming language to transform learning high school-level math topics like algebra, geometry, trigonometry, and calculus! Math Adventures with Python will show you how to harness the power of programming to keep math relevant and fun. With the aid of the Python programming language, you'll learn how to visualize solutions to a range of math problems as you use code to explore key mathematical concepts like algebra, trigonometry, matrices, and cellular automata. Once you've learned the programming basics like loops and variables, you'll write your own programs to solve equations quickly, make cool things like an interactive rainbow grid, and automate tedious tasks like factoring numbers and finding square roots. You'll learn how to write functions to draw and manipulate shapes, create oscillating sine waves, and solve equations graphically. You'll also learn how to: - Draw and transform 2D and 3D graphics with matrices - Make colorful designs like the Mandelbrot and Julia sets with complex numbers - Use recursion to create fractals like the Koch snowflake and the Sierpinski triangle - Generate virtual sheep that graze on grass and multiply autonomously - Crack secret codes using genetic algorithms As you work through the book's numerous examples and increasingly challenging exercises, you'll code your own solutions, create beautiful visualizations, and see just how much more fun math can be!*

*The fundamental mathematical tools needed to understand machine learning include linear algebra, analytic geometry, matrix decompositions, vector calculus, optimization, probability and statistics. These topics are traditionally taught in disparate courses, making it hard for data science or computer science students, or professionals, to efficiently learn the mathematics. This self-contained textbook bridges the gap between mathematical and machine learning texts, introducing the mathematical concepts with a minimum of prerequisites. It uses these concepts to derive four central machine learning methods: linear regression, principal component analysis, Gaussian mixture models and support vector machines. For students and others with a mathematical background, these derivations provide a starting point to machine learning texts. For those learning the mathematics for the first time, the methods help build intuition and practical experience with applying mathematical concepts. Every chapter includes worked examples and exercises to test understanding. Programming tutorials are offered on the book's web site.*

*Use Programming to Explore Algebra, Statistics, Calculus, and More!*

*Doing Math With Python*

*Geeky Projects for the Curious Programmer*

*Beyond the Basic Stuff with Python*

*An Illustrated Guide to Exploring Math with Code*

*Python Hacking for Math Junkies*

*Cracking Codes with Python*

Python is one of the most popular programming languages and is used in many different areas. Unlike other languages, it has a grammar familiar to people's language, so it is easy to learn and has low barriers to application. In particular, sympy, a python module introduced in this book, can represent most theories and expressions of mathematics, thus facilitating the

acquisition of concepts as well as complex calculations. This book mainly uses the sympy module of python to understand the concepts of differential and integral, and introduces various calculations of differential and integral. Derivatives and integrals are used to implicitly denote the meaning of an expression. In order to understand the implications, it is necessary to understand the calculation process of expressions. In order to understand such a meaning, various methods are used in calculus. This book introduces various techniques of calculus and the various mathematical knowledge used in its calculations using python. This course will help you understand mathematical concepts in this area as well as understand and use the python language.

This practical guide provides nearly 200 self-contained recipes to help you solve machine learning challenges you may encounter in your daily work. If you're comfortable with Python and its libraries, including pandas and scikit-learn, you'll be able to address specific problems such as loading data, handling text or numerical data, model selection, and dimensionality reduction and many other topics. Each recipe includes code that you can copy and paste into a toy dataset to ensure that it actually works. From there, you can insert, combine, or adapt the code to help construct your application. Recipes also include a discussion that explains the solution and provides meaningful context. This cookbook takes you beyond theory and concepts by providing the nuts and bolts you need to construct working machine learning applications. You'll find recipes for: Vectors, matrices, and arrays Handling numerical and categorical data, text, images, and dates and times Dimensionality reduction using feature extraction or feature selection Model evaluation and selection Linear and logical regression, trees and forests, and k-nearest neighbors Support vector machines (SVM), naïve Bayes, clustering, and neural networks Saving and loading trained models

Doing Math with Python shows you how to use Python to delve into high school-level math topics like statistics, geometry, probability, and calculus. You'll start with simple projects, like a factoring program and a quadratic-equation solver, and then create more complex projects once you've gotten the hang of things. Along the way, you'll discover new ways to explore math and gain valuable programming skills that you'll use throughout your study of math and computer science. Learn how to: -Describe your data with statistics, and visualize it with line graphs, bar charts, and scatter plots -Explore set theory and probability with programs for coin flips, dicing, and other games of chance -Solve algebra problems using Python's symbolic math functions

-Draw geometric shapes and explore fractals like the Barnsley fern, the Sierpinski triangle, and the Mandelbrot set -Write programs to find derivatives and integrate functions Creative coding challenges and applied examples help you see how you can put your new math and coding skills into practice. You'll write an inequality solver, plot gravity's effect on how far a bullet will travel, shuffle a deck of cards, estimate the area of a circle by throwing 100,000 "darts" at a board, explore the relationship between the Fibonacci sequence and the golden ratio, and more. Whether you're interested in math but have yet to dip into programming or you're a teacher looking to bring programming into the classroom, you'll find that Python makes programming easy and practical. Let Python handle the grunt work while you focus on the math. Uses Python 3

An indispensable collection of practical tips and real-world advice for tackling common Python problems and taking your code to the next level. Features interviews with high-profile Python developers who share their tips, tricks, best practices, and real-world advice gleaned from years of experience. Sharpen your Python skills as you dive deep into the Python programming language with *Serious Python*. You'll cover a range of advanced topics like multithreading and memorization, get advice from experts on things like designing APIs and dealing with databases, and learn Python internals to help you gain a deeper understanding of the language itself. Written for developers and experienced programmers, *Serious Python* brings together over 15 years of Python experience to teach you how to avoid common mistakes, write code more efficiently, and build better programs in less time. As you make your way through the book's extensive tutorials, you'll learn how to start a project and tackle topics like versioning, layouts, coding style, and automated checks. You'll learn how to package your software for distribution, optimize performance, use the right data structures, define functions efficiently, pick the right libraries, build future-proof programs, and optimize your programs down to the bytecode. You'll also learn how to:

- Make and use effective decorators and methods, including abstract, static, and class methods
- Employ Python for functional programming using generators, pure functions, and functional functions
- Extend flake8 to work with the abstract syntax tree (AST) to introduce more sophisticated automatic checks into your programs
- Apply dynamic performance analysis to identify bottlenecks in your code
- Work with relational databases and effectively manage and stream data with PostgreSQL

If you've been looking for a way to take your Python skills from good to great, *Serious Python* will help you get there. Learn from the experts and get seriously good at Python with *Serious Python*!

**Python and Matplotlib Essentials for Scientists and Engineers**

**Machine Learning with Python Cookbook**

**Invent Your Own Computer Games with Python, 4E**

**Learn Python 3 the Hard Way**

**A Primer on Scientific Programming with Python**

**Black-Belt Advice on Deployment, Scalability, Testing, and More**

**Math for Deep Learning**

*You Will Learn Python 3! Zed Shaw has perfected the world's best system for learning Python 3. Follow it and you will succeed—just like the millions of beginners Zed has taught to date! You bring the discipline, commitment, and persistence; the author supplies everything else. In Learn Python 3 the Hard Way, you'll learn Python by working through 52 brilliantly crafted exercises. Read them. Type their code precisely. (No copying and pasting!) Fix your mistakes. Watch the programs run. As you do, you'll learn how a computer works; what good programs look like; and how to read, write, and think about code. Zed then teaches you even more in 5+ hours of video where he shows you how to break, fix, and debug your code—live, as he's doing the exercises. Install a complete Python environment Organize and write code Fix and break code Basic mathematics Variables Strings and text Interact with users Work with files Looping and logic Data structures using lists and dictionaries Program design Object-oriented programming Inheritance and composition Modules, classes, and objects Python packaging Automated testing Basic game development Basic web development It'll be hard at first. But soon, you'll just get it—and that will feel great! This course will reward you for every minute you put into it. Soon, you'll know one of the world's most powerful, popular programming languages. You'll be a Python programmer. This Book Is Perfect For Total beginners with zero programming experience Junior developers who know one or two languages Returning professionals who haven't written code in years Seasoned professionals looking for a fast, simple, crash course in Python 3*

*Python is a powerful programming language that's easy to learn and fun to play with. But once you've gotten a handle on the basics, what do you do next? Python Playground is a collection of imaginative programming projects that will inspire you to use Python to make art and music, build simulations of real-world phenomena, and interact with hardware like the Arduino and Raspberry Pi. You'll learn to use common Python tools and libraries like numpy, matplotlib, and pygame to do things like:*

- Generate Spirograph-like patterns using parametric equations and the turtle module*
- Create music on your computer by simulating frequency overtones*
- Translate graphical images into ASCII art*
- Write an autostereogram program that produces 3D images hidden beneath random patterns*
- Make realistic animations with OpenGL shaders by exploring particle systems, transparency, and billboard techniques*
- Construct 3D visualizations using data from CT and MRI scans*
- Build a laser show that responds to music by hooking up your computer to an Arduino*

*Programming shouldn't be a chore. Have some solid, geeky fun with Python Playground. The projects in this book are compatible with both Python 2 and 3.*



*Invent Your Own Computer Games with Python will teach you how to make computer games using the popular Python programming language—even if you've never programmed before! Begin by building classic games like Hangman, Guess the Number, and Tic-Tac-Toe, and then work your way up to more advanced games, like a text-based treasure hunting game and an animated collision-dodging game with sound effects. Along the way, you'll learn key programming and math concepts that will help you take your game programming to the next level. Learn how to: -Combine loops, variables, and flow control statements into real working programs -Choose the right data structures for the job, such as lists, dictionaries, and tuples -Add graphics and animation to your games with the pygame module -Handle keyboard and mouse input -Program simple artificial intelligence so you can play against the computer -Use cryptography to convert text messages into secret code -Debug your programs and find common errors As you work through each game, you'll build a solid foundation in Python and an understanding of computer science fundamentals. What new game will you create with the power of Python? The projects in this book are compatible with Python 3.*

*Doing Math with Python Use Programming to Explore Algebra, Statistics, Calculus, and More! No Starch Press  
Second Edition*

*Mathematical Logic through Python*

*Applying Math with Python*

*Learn to Code by Solving Problems*

*A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code*

*Artificial Intelligence with Python*

*Powerful Object-Oriented Programming*

This book presents computer programming as a key method for solving mathematical problems. There are two versions of the book, one for MATLAB and one for Python. The book was inspired by the Springer book TCSE 6: A Primer on Scientific Programming with Python (by Langtangen), but the style is more accessible and concise, in keeping with the needs of engineering students. The book outlines the shortest possible path from no previous experience with programming to a set of skills that allows the students to write simple programs for solving common mathematical problems with numerical methods in engineering and science courses. The emphasis is on generic algorithms, clean design of programs, use of functions, and automatic tests for verification.

Corrected and revised with additional material. Computers are wonderful things. They improve our lives and produce works of art. Information and communication are free and available in ways that were undreamed of by many a generation ago. And yet computers inspire fear when they are used improperly or misunderstood. Everyone uses computers today. The typical student uses at least three: a portable notebook computer, a tablet, and a cell phone. None

of these even existed before this year's college graduating class was born. So a reasonable amount of computer literacy is expected of everyone these days. Though inspired by mathematics, much of the mathematical beauty of computation is largely ignored by both elementary computer science and introductory programming books. This book is designed to help math junkies - anyone who likes math, studies math, or uses math in their daily life - learn about computation. The emphasis is on algorithms. It is appropriate for students with no prior programming experience as well as professional scientists. Contents: Python programs, iPython notebooks, expressions, statements, types, lists, arrays, functions, classes, plotting, list comprehension, recursion. Applications include linear systems, computational geometry, root finding, interpolation, polynomial least squares, discrete systems, differential equations, principal component analysis, singular value decomposition, fractals, chaos, and satellite orbit propagation. Also covers numpy, plotting with pyplot, and basics maps with basemap. Now includes more than 200 exercises for students. Target audience: undergraduate math and science students entering their upper level curriculum. Calculus and some linear algebra background is helpful; a review of linear algebra is included as an appendix. No prior programming experience is expected

The book serves as a first introduction to computer programming of scientific applications, using the high-level Python language. The exposition is example and problem-oriented, where the applications are taken from mathematics, numerical calculus, statistics, physics, biology and finance. The book teaches "Matlab-style" and procedural programming as well as object-oriented programming. High school mathematics is a required background and it is advantageous to study classical and numerical one-variable calculus in parallel with reading this book. Besides learning how to program computers, the reader will also learn how to solve mathematical problems, arising in various branches of science and engineering, with the aid of numerical methods and programming. By blending programming, mathematics and scientific applications, the book lays a solid foundation for practicing computational science. From the reviews: Langtangen ... does an excellent job of introducing programming as a set of skills in problem solving. He guides the reader into thinking properly about producing program logic and data structures for modeling real-world problems using objects and functions and embracing the object-oriented paradigm. ... Summing Up: Highly recommended. F. H. Wild III, Choice, Vol. 47 (8), April 2010 Those of us who have learned scientific programming in Python ' on the streets ' could be a little jealous of students who have the opportunity to take a course out of Langtangen ' s Primer. " John D. Cook, The Mathematical Association of America, September 2011 This book goes through Python in particular, and programming in general, via tasks that scientists will likely perform. It contains valuable information for students new to scientific computing and would be the perfect bridge between an introduction to programming and an advanced course on numerical methods or computational science. Alex Small, IEEE, CiSE Vol. 14 (2), March /April 2012 " This fourth edition is

a wonderful, inclusive textbook that covers pretty much everything one needs to know to go from zero to fairly sophisticated scientific programming in Python... ” Joan Horvath, Computing Reviews, March 2015

Impractical Python Projects is a collection of fun and educational projects designed to entertain programmers while enhancing their Python skills. It picks up where the complete beginner books leave off, expanding on existing concepts and introducing new tools that you'll use every day. And to keep things interesting, each project includes a zany twist featuring historical incidents, pop culture references, and literary allusions. You'll flex your problem-solving skills and employ Python's many useful libraries to do things like: - Help James Bond crack a high-tech safe with a hill-climbing algorithm - Write haiku poems using Markov Chain Analysis - Use genetic algorithms to breed a race of gigantic rats - Crack the world's most successful military cipher using cryptanalysis - Derive the anagram, "I am Lord Voldemort" using linguistical sieves - Plan your parents' secure retirement with Monte Carlo simulation - Save the sorceress Zatanna from a stabby death using palindromes - Model the Milky Way and calculate our odds of detecting alien civilizations - Help the world's smartest woman win the Monty Hall problem argument - Reveal Jupiter's Great Red Spot using optical stacking - Save the head of Mary, Queen of Scots with steganography - Foil corporate security with invisible electronic ink Simulate volcanoes, map Mars, and more, all while gaining valuable experience using free modules like Tkinter, matplotlib, Cprofile, Pylint, Pygame, Pillow, and Python-Docx. Whether you're looking to pick up some new Python skills or just need a pick-me-up, you'll find endless educational, geeky fun with Impractical Python Projects.

Python in a Nutshell

Best Practices for Writing Clean Code

Practical Solutions from Preprocessing to Deep Learning

A Python Programming Primer

What You Need to Know to Understand Neural Networks

Coding in Python and Elements of Discrete Mathematics

Learning Python

*Get a comprehensive, in-depth introduction to the core Python language with this hands-on book. Based on author Mark Lutz's popular training course, this updated fifth edition will help you quickly write efficient, high-quality code with Python. It's an ideal way to begin, whether you're new to programming or a professional developer versed in other languages. Complete with quizzes, exercises, and helpful illustrations, this easy-to-follow, self-paced tutorial gets you started with both Python 2.7 and 3.3—the latest releases in the 3.X and 2.X lines—plus all other releases in common use today. You'll also learn some advanced language features that recently have become more common in Python code. Explore Python's major built-in object types such as numbers, lists, and dictionaries Create and process objects with Python statements, and learn Python's general syntax model Use functions to avoid code redundancy and package code for reuse*

*Organize statements, functions, and other tools into larger components with modules Dive into classes: Python's object-oriented programming tool for structuring code Write large programs with Python's exception-handling model and development tools Learn advanced Python tools, including decorators, descriptors, metaclasses, and Unicode processing*

*Game design requires many skills including imagination, problem solving, communication, and teamwork. These characteristics make it a natural fit for the Maker movement. From board games to video games, this exciting title introduces readers to the essential basics of game design including game components and systems, prototype design, play testing, and the steps in the iterative design process. "Makers and Shakers" sidebars introduce readers to some of the world's greatest game designers and innovators. The title also includes engaging, step-by-step Maker projects to put their game design skills to work*

*A unique approach to mathematical logic where students implement the underlying concepts and proofs in the Python programming language.*

*Learn how to program in Python while making and breaking ciphers—algorithms used to create and send secret messages! After a crash course in Python programming basics, you'll learn to make, test, and hack programs that encrypt text with classical ciphers like the transposition cipher and Vigenère cipher. You'll begin with simple programs for the reverse and Caesar ciphers and then work your way up to public key cryptography, the type of encryption used to secure today's online transactions, including digital signatures, email, and Bitcoin. Each program includes the full code and a line-by-line explanation of how things work. By the end of the book, you'll have learned how to code in Python and you'll have the clever programs to prove it! You'll also learn how to: - Combine loops, variables, and flow control statements into real working programs - Use dictionary files to instantly detect whether decrypted messages are valid English or gibberish - Create test programs to make sure that your code encrypts and decrypts correctly - Code (and hack!) a working example of the affine cipher, which uses modular arithmetic to encrypt a message - Break ciphers with techniques such as brute-force and frequency analysis There's no better way to learn to code than to play with real programs. Cracking Codes with Python makes the learning fun!*

*Math for Programmers*

*Neighborhood Cleanup*

*Matematicas Y Programacion En Python*

*Exploring Math Through Computer Programming*

*3D graphics, machine learning, and simulations with Python*

*Calculus Story I with Python*

*Serious Python*

Power up your Python with object-oriented programming and learn how to write powerful, efficient, and re-usable code. Object-oriented Python is an intuitive and thorough guide to mastering object-oriented programming from the ground up. You'll cover the basics of building classes and creating objects, and put theory into practice using the pygame package with clear examples that help visualize the object-oriented style. You'll explore the key concepts of object-oriented programming — encapsulation, polymorphism, and inheritance — and learn not just how to code with objects, but the absolute best practices for doing so. In the end, you'll bring it all together by building a complex video game, complete with full animations and sounds. The book covers two

functional Python code packages that will speed up development of graphical user interface (GUI) programs in Python. Math for Deep Learning provides the essential math you need to understand deep learning discussions, explore more complex implementations, and better use the deep learning toolkits. With Math for Deep Learning, you'll learn the essential mathematics used by and as a background for deep learning. You'll work through Python examples to learn key deep learning related topics: probability, statistics, linear algebra, differential calculus, and matrix calculus as well as how to implement data flow in a neural network, backpropagation, and gradient descent. You'll also use Python to work through the mathematics that underlies those algorithms and even build a fully-functional neural network. In addition you'll find coverage of gradient descent including variants commonly used by the deep learning community: SGD, Adam, RMSprop, and Adagrad/Adadelta.

This open access book offers an initial introduction to programming for scientific and computational applications using the Python programming language. The presentation style is compact and example-based, making it suitable for students and researchers with little or no prior experience in programming. The book uses relevant examples from mathematics and the natural sciences to present programming as a practical toolbox that can quickly enable readers to write their own programs for data processing and mathematical modeling. These tools include file reading, plotting, simple text analysis, and using NumPy for numerical computations, which are fundamental building blocks of all programs in data science and computational science. At the same time, readers are introduced to the fundamental concepts of programming, including variables, functions, loops, classes, and object-oriented programming. Accordingly, the book provides a sound basis for further computer science and programming studies.

"This fantastic and deep book about how to use Sage for learning and doing mathematics at all levels perfectly complements existing Sage documentation. It is filled with many carefully thought through examples and exercises, and great care has been taken to put computational functionality into proper mathematical context. Flip to almost any random page in this amazing book and you will learn how to play with and visualize some beautiful part of mathematics." --- William A. Stein, CEO, SageMath, an associate professor of mathematics, University of Washington SageMath, or Sage for short, is an open-source mathematical software system based on the Python language and developed by an international community comprising hundreds of teachers and researchers, whose aim is to provide an alternative to the commercial products Magma, Maple, Mathematica, and MATLAB®. To achieve this, Sage relies on many open-source programs, including GAP, Maxima, PARI, and various scientific libraries for Python, to which thousands of new functions have been added. Sage is freely available and is supported by all modern operating systems. Sage provides a wonderful scientific and graphical calculator for high school students, and it efficiently supports undergraduates in their computations in analysis, linear algebra, calculus, etc. For graduate students, researchers, and engineers in various mathematical specialties, Sage provides the most recent algorithms and tools, which is why several universities around the world already use Sage at the undergraduate level.

Python Playground

150 Questions with Solutions for PYTHON PROGRAMMING

Playful Programming Activities to Make You Smarter  
Maker Projects for Kids Who Love Games  
Master OOP by Building Games and GUIs  
Object-Oriented Python

*A new kind of math book! Explore math topics from arithmetic to calculus by creating your own graphing and solving tools using Python. Create 2D and 3D graphics, harmonograph and spirograph designs, and fractals in this interactive and visual exploration of mathematics. "A great resource to play with Math and Python via the turtle module, solving equations numerically and 3D graphics via Pi3D." - Amit Saha, author of Doing Math With Python Imagine learning math and Python programming at the same time! You'll learn to use loops, variables, functions, conditionals and lists and apply them to all your math problems. No previous computer experience is required.*

*Lillian stared at the closed door through which both her father and Mrs. Bolstreath had passed, and then looked at Dan, sitting somewhat disconsolately at the disordered dinner-table. She was a delicately pretty girl of a fair fragile type, not yet twenty years of age, and resembled a shepherdess of Dresden china in her dainty perfection. With her pale golden hair, and rose-leaf complexion; arrayed in a simple white silk frock with snowy pearls round her slender neck, she looked like a wreath of faint mist. At least Dan fancifully thought so, as he stole a glance at her frail beauty, or perhaps she was more like a silver-point drawing, exquisitely fine. But whatever image love might find to express her loveliness, Dan knew in his hot passion that she was the one girl in the world for him. Lillian Halliday was a much better name for her than Lillian Moon. Dan himself was tall and slim, dark and virile, with a clear-cut, clean-shaven face suggestive of strength and activity. His bronzed complexion suggested an open-air life, while the eagle look in his dark eyes was that new vast-distance expression rapidly being acquired by those who devote themselves to aviation. No one could deny Dan's good looks or clean life or daring nature, and he was all that a girl could desire in the way of a fairy prince. But fathers do not approve of fairy princes unless they come laden with jewels and gold. To bring such to Lillian was rather like taking coals to Newcastle since her father was so wealthy; but much desires more, and Sir Charles wanted a rich son-in-law. Dan could not supply this particular adjective, and therefore--as he would have put it in the newest slang of the newest*

profession--was out of the fly. Not that he intended to be, in spite of Sir Charles, since love can laugh at stern fathers as easily as at bolts and bars. And all this time Lillian stared at the door, and then at Dan, and then at her plate, putting two and two together. But in spite of her feminine intuition, she could not make four, and turned to her lover--for that Dan was, and a declared lover too--for an explanation.

**BRIDGE THE GAP BETWEEN NOVICE AND PROFESSIONAL** You've completed a basic Python programming tutorial or finished Al Sweigart's bestseller, *Automate the Boring Stuff with Python*. What's the next step toward becoming a capable, confident software developer? Welcome to *Beyond the Basic Stuff with Python*. More than a mere collection of advanced syntax and masterful tips for writing clean code, you'll learn how to advance your Python programming skills by using the command line and other professional tools like code formatters, type checkers, linters, and version control. Sweigart takes you through best practices for setting up your development environment, naming variables, and improving readability, then tackles documentation, organization and performance measurement, as well as object-oriented design and the Big-O algorithm analysis commonly used in coding interviews. The skills you learn will boost your ability to program--not just in Python but in any language. You'll learn:

- Coding style, and how to use Python's Black auto-formatting tool for cleaner code
- Common sources of bugs, and how to detect them with static analyzers
- How to structure the files in your code projects with the Cookiecutter template tool
- Functional programming techniques like lambda and higher-order functions
- How to profile the speed of your code with Python's built-in timeit and cProfile modules
- The computer science behind Big-O algorithm analysis
- How to make your comments and docstrings informative, and how often to write them
- How to create classes in object-oriented programming, and why they're used to organize code

Toward the end of the book you'll read a detailed source-code breakdown of two classic command-line games, the Tower of Hanoi (a logic puzzle) and Four-in-a-Row (a two-player tile-dropping game), and a breakdown of how their code follows the book's best practices. You'll test your skills by implementing the program yourself. Of course, no single book can make you a professional software developer. But *Beyond the Basic Stuff with Python* will get you further down that path and make you a better programmer, as you learn to write readable code that's easy to debug and perfectly Pythonic

**Requirements: Covers Python 3.6 and higher**

**Build real-world Artificial Intelligence applications with Python to intelligently interact with the world around you**

**About This Book** Step into the amazing world of intelligent apps using this

*comprehensive guide Enter the world of Artificial Intelligence, explore it, and create your own applications Work through simple yet insightful examples that will get you up and running with Artificial Intelligence in no time Who This Book Is For This book is for Python developers who want to build real-world Artificial Intelligence applications. This book is friendly to Python beginners, but being familiar with Python would be useful to play around with the code. It will also be useful for experienced Python programmers who are looking to use Artificial Intelligence techniques in their existing technology stacks. What You Will Learn Realize different classification and regression techniques Understand the concept of clustering and how to use it to automatically segment data See how to build an intelligent recommender system Understand logic programming and how to use it Build automatic speech recognition systems Understand the basics of heuristic search and genetic programming Develop games using Artificial Intelligence Learn how reinforcement learning works Discover how to build intelligent applications centered on images, text, and time series data See how to use deep learning algorithms and build applications based on it In Detail Artificial Intelligence is becoming increasingly relevant in the modern world where everything is driven by technology and data. It is used extensively across many fields such as search engines, image recognition, robotics, finance, and so on. We will explore various real-world scenarios in this book and you'll learn about various algorithms that can be used to build Artificial Intelligence applications. During the course of this book, you will find out how to make informed decisions about what algorithms to use in a given context. Starting from the basics of Artificial Intelligence, you will learn how to develop various building blocks using different data mining techniques. You will see how to implement different algorithms to get the best possible results, and will understand how to apply them to real-world scenarios. If you want to add an intelligence layer to any application that's based on images, text, stock market, or some other form of data, this exciting book on Artificial Intelligence will definitely be your guide! Style and approach This highly practical book will show you how to implement Artificial Intelligence. The book provides multiple examples enabling you to create smart applications to meet the needs of your organization. In every chapter, we explain an algorithm, implement it, and then build a smart application.*

*Scientific Computation*

*Automate the Boring Stuff with Python, 2nd Edition*

*Hacking Math Class with Python*



## **Math Adventures with Python**

### **Mathematics for Machine Learning**

*In Math for Programmers you'll explore important mathematical concepts through hands-on coding. Filled with graphics and more than 300 exercises and mini-projects, this book unlocks the door to interesting-and lucrative!-careers in some of today's hottest fields. As you tackle the basics of linear algebra, calculus, and machine learning, you'll master the key Python libraries used to turn them into real-world software applications. Summary To score a job in data science, machine learning, computer graphics, and cryptography, you need to bring strong math skills to the party. Math for Programmers teaches the math you need for these hot careers, concentrating on what you need to know as a developer. Filled with lots of helpful graphics and more than 200 exercises and mini-projects, this book unlocks the door to interesting-and lucrative!-careers in some of today's hottest programming fields. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the technology Skip the mathematical jargon: This one-of-a-kind book uses Python to teach the math you need to build games, simulations, 3D graphics, and machine learning algorithms. Discover how algebra and calculus come alive when you see them in code! About the book In Math for Programmers you'll explore important mathematical concepts through hands-on coding. Filled with graphics and more than 300 exercises and mini-projects, this book unlocks the door to interesting-and lucrative!-careers in some of today's hottest fields. As you tackle the basics of linear algebra, calculus, and machine learning, you'll master the key Python libraries used to turn them into real-world software applications. What's inside Vector geometry for computer graphics Matrices and linear transformations Core concepts from calculus Simulation and optimization Image and audio processing Machine learning algorithms for regression and classification About the reader For programmers with basic skills in algebra. About the author Paul Orland is a programmer, software entrepreneur, and math enthusiast. He is co-founder of Tachyus, a start-up building predictive analytics software for the energy industry. You can find him online at [www.paulor.land](http://www.paulor.land). Table of Contents 1 Learning math with code PART I - VECTORS AND GRAPHICS 2 Drawing with*

2D vectors 3 Ascending to the 3D world 4 Transforming vectors and graphics 5 Computing transformations with matrices 6 Generalizing to higher dimensions 7 Solving systems of linear equations PART 2 - CALCULUS AND PHYSICAL SIMULATION 8 Understanding rates of change 9 Simulating moving objects 10 Working with symbolic expressions 11 Simulating force fields 12 Optimizing a physical system 13 Analyzing sound waves with a Fourier series PART 3 - MACHINE LEARNING APPLICATIONS 14 Fitting functions to data 15 Classifying data with logistic regression 16 Training neural networks

The second edition of this best-selling Python book (over 500,000 copies sold!) uses Python 3 to teach even the technically uninclined how to write programs that do in minutes what would take hours to do by hand. There is no prior programming experience required and the book is loved by liberal arts majors and geeks alike. If you've ever spent hours renaming files or updating hundreds of spreadsheet cells, you know how tedious tasks like these can be. But what if you could have your computer do them for you? In this fully revised second edition of the best-selling classic *Automate the Boring Stuff with Python*, you'll learn how to use Python to write programs that do in minutes what would take you hours to do by hand--no prior programming experience required. You'll learn the basics of Python and explore Python's rich library of modules for performing specific tasks, like scraping data off websites, reading PDF and Word documents, and automating clicking and typing tasks. The second edition of this international fan favorite includes a brand-new chapter on input validation, as well as tutorials on automating Gmail and Google Sheets, plus tips on automatically updating CSV files. You'll learn how to create programs that effortlessly perform useful feats of automation to:

- Search for text in a file or across multiple files
- Create, update, move, and rename files and folders
- Search the Web and download online content
- Update and format data in Excel spreadsheets of any size
- Split, merge, watermark, and encrypt PDFs
- Send email responses and text notifications
- Fill out online forms

Step-by-step instructions walk you through each program, and updated practice projects at the end of each chapter challenge you to improve those programs and use your newfound skills to automate similar tasks. Don't spend your time doing work a well-trained monkey could do. Even if you've

*never written a line of code, you can make your computer do the grunt work. Learn how in Automate the Boring Stuff with Python, 2nd Edition.*

*This book provides an introduction to the core features of the Python programming language and Matplotlib plotting routings for scientists and engineers (or students of either discipline) who want to use Python™ to analyse data, simulate physical processes, and render publication-quality plots. No previous programming experience is needed before reading the first page. Readers will learn the core features of the Python programming language in under a day. They will be able to immediately use Python to implement codes that solve their own problems and make beautiful plots and animations. Python code is extremely fast to prototype, allowing users to achieve results quickly and accurately. The examples within the book are available for download at <http://pythonessentials.com>. Python and Matplotlib Essentials for Scientists and Engineers is accessible for motivated high-school students, but will likely be most useful for undergraduate and graduate students as well as working professionals who have some background with the basic mathematical concepts. This book is intended for technical people who want to get things done.*