programming languages inspired by python

programming languages inspired by python have played a significant role in shaping modern software development. Python's simplicity, readability, and versatility have influenced numerous programming languages designed to enhance developer productivity and maintain code clarity. This article explores various programming languages that draw inspiration from Python's design philosophy, syntax, and features. These languages often emphasize ease of use, dynamic typing, and clean code structure, making them popular choices for developers across different domains. Understanding these languages provides insight into how Python's impact extends beyond its own ecosystem, fostering innovation in language design. This article covers key languages inspired by Python, their unique characteristics, and how they integrate Python's principles to meet specific programming needs.

Languages Directly Influenced by Python

Several programming languages have been directly influenced by Python, adopting its core principles such as readability, simplicity, and dynamic typing. These languages often aim to build upon Python's strengths while addressing specific shortcomings or targeting niche applications.

Julia

Julia is a high-level, high-performance programming language designed for technical computing. It draws inspiration from Python's readable syntax and dynamic typing but focuses heavily on numerical and scientific computing performance. Julia's use of just-in-time (JIT) compilation allows it to achieve speeds comparable to statically-typed languages like C, while maintaining Python-like ease of use.

Groovy

Groovy is a dynamic language for the Java Virtual Machine (JVM) that adopts Python's flexible syntax and

dynamic typing. It integrates seamlessly with Java libraries and emphasizes concise and readable code, similar to Python. Groovy supports scripting, domain-specific language creation, and rapid prototyping, making it a popular choice for developers seeking Python's simplicity on the JVM.

Ruby

Though predating Python in some aspects, Ruby shares many design philosophies with Python, including a focus on developer happiness, readability, and elegance. Ruby's syntax and dynamic features reflect similar goals, making it a language often compared to Python in terms of ease of learning and expressiveness.

Languages with Python-Like Syntax

Some programming languages emulate Python's clear and concise syntax to improve developer experience. These languages aim to provide an intuitive coding style that reduces complexity and enhances maintainability.

Cobra

Cobra is a statically-typed language that combines Python's readable syntax with advanced features such as contracts, unit tests, and static analysis. It targets application development where both performance and clarity are essential. Cobra's syntax and indentation-based structure closely resemble Python, allowing developers familiar with Python to adapt quickly.

CoffeeScript

CoffeeScript is a language that compiles into JavaScript but uses a Python-inspired syntax to simplify JavaScript development. By adopting Python's indentation style and concise expressions, CoffeeScript reduces boilerplate code and enhances readability, making it easier to write and maintain client-side scripts.

Swift

Apple's Swift language incorporates some Python-like features, particularly in its emphasis on readable, expressive syntax. Swift's design promotes clarity and safety while supporting powerful modern programming constructs. Although Swift is statically typed, its syntax aims to reduce verbosity, a trait admired in Python.

Functional and Scripting Languages Inspired by Python

Python's versatility as a scripting language and its support for multiple programming paradigms have inspired several languages that blend functional programming concepts with Python's approachable syntax.

Elvish

Elvish is a modern shell language that combines the scripting ease of Python with functional programming features. It uses a clean syntax and powerful data structures, aiming to replace traditional Unix shells. Elvish's design prioritizes readability and expressiveness, reflecting Python's influence on scripting environments.

Nim

Nim is a statically-typed compiled language that offers Python-like syntax with the performance of C. Nim supports multiple programming paradigms, including imperative, functional, and object-oriented programming. Its clean syntax and compile-time features demonstrate Python's impact on newer, performance-oriented languages.

Lua

While Lua predates Python, recent versions of Lua have incorporated features inspired by Python's simplicity and flexibility. Lua is widely used for scripting in embedded systems and games. Its lightweight nature combined with Python-inspired usability makes it a popular choice for rapid development.

Impact of Python's Philosophy on Modern Language Design

Python's design philosophy, characterized by readability, simplicity, and explicitness, has left a lasting imprint on modern programming languages. This impact extends beyond syntax, influencing language features, tooling, and community practices.

Readability and Explicitness

Languages inspired by Python often emphasize readable code that is easy to understand and maintain. This focus on explicitness reduces errors and improves collaboration among developers. The use of indentation-based blocks and clear naming conventions are common traits inherited from Python's philosophy.

Dynamic Typing and Flexibility

Dynamic typing, a hallmark of Python, allows for rapid prototyping and flexible code development. Many languages inspired by Python adopt dynamic or gradual typing systems to balance flexibility with safety. These typing strategies enable developers to write expressive code without sacrificing performance or reliability.

Community and Ecosystem Influence

Python's extensive standard library and vibrant community have set standards for language ecosystems. Inspired languages often strive to provide rich libraries, package managers, and community-driven resources. This holistic approach to language design fosters innovation and widespread adoption.

Key Features Adopted from Python

- Indentation-based syntax for code blocks
- Emphasis on simplicity and minimalism
- Support for multiple programming paradigms
- Dynamic and gradual typing systems
- Comprehensive standard libraries and tooling
- Interactive interpreters and REPL environments

Frequently Asked Questions

What are some programming languages inspired by Python?

Languages such as Julia, Ruby, and Cobra have drawn inspiration from Python's syntax and design philosophy.

How does Julia draw inspiration from Python?

Julia adopts Python's readability and simplicity while focusing on high-performance numerical computing

and scientific applications.

Is Ruby influenced by Python in any way?

Yes, Ruby shares Python's emphasis on clean, readable code and developer happiness, though it has a more object-oriented approach.

What features of Python have influenced newer programming languages?

Features like dynamic typing, indentation-based syntax, extensive standard libraries, and emphasis on code readability have influenced many modern languages.

Are there any statically typed languages inspired by Python?

Yes, languages like Cobra combine Python's readability with static typing and contracts for enhanced reliability.

How does Cobra relate to Python?

Cobra is a programming language that incorporates Python-like syntax but adds static typing and design-by-contract features.

Have any domain-specific languages been inspired by Python?

Some domain-specific languages for data science and machine learning adopt Python's syntax style to make them more accessible to Python programmers.

Does Python influence the design of scripting languages?

Yes, many scripting languages take cues from Python's simplicity and ease of use to improve developer productivity.

What makes Python's design appealing to other language creators?

Python's clean syntax, readability, simplicity, and extensive ecosystem make it a popular model for new language designs.

Are there languages that combine Python's syntax with other programming paradigms?

Yes, languages like Julia combine Python-like syntax with multiple paradigms such as functional and

imperative programming, aiming for performance and expressiveness.

Additional Resources

1. *Pythonic Paradigms: Mastering Elegant Code*
This book delves into writing clean, efficient, and readable Python code by embracing the Pythonic way of programming. It covers idiomatic constructs, best practices, and design patterns that make Python stand out. Readers will learn how to write code that is not only functional but also expressive and maintainable.

2. *MicroPython for Embedded Systems: Programming the Future*
Focused on MicroPython, this book guides readers through using Python on microcontrollers and embedded devices. It covers hardware interfacing, real-time programming concepts, and practical projects. Ideal for hobbyists and professionals wanting to bring Python to IoT and embedded applications.

3. *PyDSL: Designing Domain-Specific Languages with Python*
Explore the art of creating domain-specific languages (DSLs) using Python as a host language. This book explains parsing, language design principles, and implementation techniques. Readers will gain skills to build custom mini-languages that solve specialized problems efficiently.

4. *Async Python: Concurrency and Parallelism Simplified*
This book unpacks asynchronous programming in Python, explaining the async/await syntax and event-driven design. It provides practical examples of concurrency, parallelism, and managing I/O-bound operations. Developers will learn to build scalable and responsive applications with ease.

5. *PyScript Revolution: Programming Python in the Browser*
Discover how PyScript enables running Python directly in web browsers. The book covers setup, integration with HTML/CSS, and creating interactive web applications. It's a guide for Python developers interested in front-end development without switching languages.

6. *Python Metaprogramming: Code That Writes Code*
Delve into advanced Python features like decorators, metaclasses, and dynamic code generation. This book teaches how to write flexible and reusable code by leveraging Python's metaprogramming capabilities. It's suited for experienced developers aiming to deepen their understanding of Python internals.

7. *Data Science with Pythonic Tools*
This comprehensive guide introduces Python libraries and frameworks essential for data science, including pandas, NumPy, and scikit-learn. It combines theoretical concepts with hands-on projects in data analysis and machine learning. Perfect for beginners and intermediate programmers seeking practical skills.

8. *Functional Python: Embracing Immutability and Higher-Order Functions*
Explore functional programming concepts implemented in Python, such as immutability, pure functions, and function composition. The book shows how to write concise, bug-resistant code by adopting functional

techniques. It's a valuable resource for developers looking to expand their programming paradigms.

9. *Python for Automation: Scripting Your Workflow*
This practical book focuses on using Python to automate repetitive tasks, from file handling to web scraping and system administration. It provides step-by-step instructions and real-world examples to boost productivity. Ideal for professionals who want to streamline their daily workflows using Python.

Programming Languages Inspired By Python

programming languages inspired by python: Raspberry Pi Supercomputing and Scientific Programming Ashwin Pajankar, 2017-05-25 Build an inexpensive cluster of multiple Raspberry Pi computers and install all the required libraries to write parallel and scientific programs in Python 3. This book covers setting up your Raspberry Pis, installing the necessary software, and making a cluster of multiple Pis. Once the cluster is built, its power has to be exploited by means of programs to run on it. So, Raspberry Pi Supercomputing and Scientific Programming teaches you to code the cluster with the MPI4PY library of Python 3. Along the way, you will learn the concepts of the Message Passing Interface (MPI) standards and will explore the fundamentals of parallel programming on your inexpensive cluster. This will make this book a great starting point for supercomputing enthusiasts who want to get started with parallel programming. The book finishes with details of symbolic mathematics and scientific and numerical programming in Python, using SymPi, SciPy, NumPy, and Matplotlib. You'll see how to process signals and images, carry out calculations using linear algebra, and visualize your results, all using Python code. With the power of a Raspberry Pi supercomputer at your fingertips, data-intensive scientific programming becomes a reality at home. What You Will Learn Discover the essentials of supercomputing Build a low-cost cluster of Raspberry Pis at home Harness the power of parallel programming and the Message Passing Interface (MPI) Use your Raspberry Pi for symbolic, numerical, and scientific programming Who This Book Is For Python 3 developers who seek the knowledge of parallel programming, Raspberry Pi enthusiasts, researchers, and the scientific Python community.

programming languages inspired by python: Programming Languages: Concepts and Implementation Saverio Perugini, 2021-12-02 Programming Languages: Concepts and Implementation teaches language concepts from two complementary perspectives: implementation and paradigms. It covers the implementation of concepts through the incremental construction of a progressive series of interpreters in Python, and Racket Scheme, for purposes of its combined simplicity and power, and assessing the differences in the resulting languages.

programming languages inspired by python: Programming Languages: Principles and Paradigms Maurizio Gabbrielli, Simone Martini, 2023-10-14 This textbook is a thorough, up-to-date introduction to the principles and techniques that guide the design and implementation of modern programming languages. The goal of the book is to provide the basis for a critical understanding of most modern programming languages. Thus, rather than focusing on a specific language, the book identifies the most important principles shared by large classes of languages. The notion of 'abstract

machine’ is a unifying concept that helps to maintain an accurate and elementary treatment. The book introduces, analyses in depth, and compares the imperative, object-oriented, functional, logic, concurrent, constraint-based, and service-oriented programming paradigms. All material coming from the first English edition has been updated and extended, clarifying some tricky points, and discussing newer programming languages. This second edition contains new chapters dedicated to constraint, concurrent, and service-oriented programming. Topics and features: Requires familiarity with one programming language is a prerequisite Provides a chapter on history offering context for most of the constructs in use today Presents an elementary account of semantical approaches and of computability Introduces new examples in modern programming languages like Python or Scala Offers a chapter that opens a perspective on applications in artificial intelligence Conceived as a university textbook, this unique volume will also be suitable for IT specialists who want to deepen their knowledge of the mechanisms behind the languages they use. The choice of themes and the presentation style are largely influenced by the experience of teaching the content as part of a bachelor's degree in computer science.

programming languages inspired by python: ,

programming languages inspired by python: *Programming in Python* Pooja Sharma, 2020-04-09 An interactive way to introduce the world of Python Programming KEY FEATURES Detailed comparisons and differentiation of python language from other most popular languages C/C++/Java. Authentic and extensive set of programming illustrations in every chapter of the book. Broad study on all the programming constructs of the python programming language such as native data types, looping, decision making, exception handling, file handling etc. Broad study of Python Object Oriented Programming features with illustrations. Numerous review questions and exercises at the end of every chapter. DESCRIPTION This Book is meant for wide range of readers who wish to learn the basics of Python programming language. It can be helpful for students, programmers, researchers, and software developers. The basic concepts of python programming are dealt in detail. The various concepts of python language such as object-oriented features, operators, native data types, control structures, functions, exception handling, file handling, etc are discussed in detail with the authentic programming illustration of each. presently, python programming is a hot topic among academicianÕs researchers, and program developers. As a result, the book is designed to give an in-depth knowledge of programming in python. This book can be used as handbook as well as a guide for students of all computer science stream at any grade beginning from 10+1 to Research in PhD. To conclude, we hope that the readers will find this book a helpful guide and valuable source of information about python programming. WHAT WILL YOU LEARN Python Data Types, Input Output Operators and Expressions Control Structures Python Functions, Modules Exception Handling File Management, Classes and Objects Inheritance, Python Operator Overloading Ê WHO THIS BOOK IS FOR Students, Programmers, researchers, and software developers who wish to learn the basics of Python programming language. Ê Table of Contents 1. Introduction to Python Language 2. Python Data Types and Input Output 3. Operators and Expressions 4. Control Structures 5. Python Native Data Types 6. Python Functions 7. Python Modules 8. Exception Handling 9. File Management in Python 10. Classes and Objects 11. Inheritance 12. Python Operator Overloading

programming languages inspired by python: *Blockchain Quick Start Guide* Xun (Brian) Wu, Weimin Sun, 2018-12-27 Learn quick and effective techniques to get up and running with building blockchain including Ethereum and Hyperledger Fabric. Key FeaturesUnderstand the key concepts of decentralized applications and consensus algorithmsLearn key concepts of Ethereum and Solidity programmingPractical guide to get started with build efficient Blockchain applications with Ethereum and HyperledgerBook Description Blockchain is a technology that powers the development of decentralized applications.This technology allows the construction of a network with no single control that enables participants to make contributions to and receive benefits from the network directly. This book will give you a thorough overview of blockchain and explain how a blockchain works.You will begin by going through various blockchain consensus mechanisms and

cryptographic hash functions. You will then learn the fundamentals of programming in Solidity – the defacto language for developing decentralize, applications in Ethereum. After that, you will set up an Ethereum development environment and develop, package, build, and test campaign-decentralized applications.The book also shows you how to set up Hyperledger composer tools, analyze business scenarios, design business models, and write a chain code. Finally, you will get a glimpse of how blockchain is actually used in different real-world domains. By the end of this guide, you will be comfortable working with basic blockchain frameworks, and develop secure, decentralized applications in a hassle-free manner. What you will learnUnderstand how blockchain hashing worksWrite and test a smart contract using SolidityDevelop and test a decentralized applicationBuild and test your application using Hyperledger FabricImplement business network using Hyperledger ComposerTest and interact with business network applicationsWho this book is for The book is for developers, analysts, or anyone looking to learn about Blockchain in a quick and easy manner.

programming languages inspired by python: Blockchain Development with Hyperledger Salman A. Baset, Luc Desrosiers, Nitin Gaur, Petr Novotny, Anthony O'Dowd, Venkatraman Ramakrishna, Weimin Sun, Xun (Brian) Wu, 2019-03-26 Learn quick and effective techniques for developing blockchain-based distributed ledgers with ease Key FeaturesDiscover why blockchain is a game changer in the technology landscapeSet up blockchain networks using Hyperledger FabricWrite smart contracts at speed with Hyperledger ComposerBook Description Blockchain and Hyperledger are open source technologies that power the development of decentralized applications. This Learning Path is your helpful reference for exploring and building blockchain networks using Ethereum, Hyperledger Fabric, and Hyperledger Composer. Blockchain Development with Hyperledger will start off by giving you an overview of blockchain and demonstrating how you can set up an Ethereum development environment for developing, packaging, building, and testing campaign-decentralized applications. You'll then explore the de facto language Solidity, which you can use to develop decentralized applications in Ethereum. Following this, you'll be able to configure Hyperledger Fabric and use it to build private blockchain networks and applications that connect to them. Toward the later chapters, you'll learn how to design and launch a network, and even implement smart contracts in chain code. By the end of this Learning Path, you'll be able to build and deploy your own decentralized applications by addressing the key pain points encountered in the blockchain life cycle. This Learning Path includes content from the following Packt products: Blockchain Quick Start Guide by Xun (Brian) Wu and Weimin SunHands-On Blockchain with Hyperledger by Nitin Gaur et al.What you will learnUnderstand why decentralized applications are necessaryDevelop and test a decentralized application with Hyperledger Fabric and Hyperledger ComposerWrite and test a smart contract using SolidityDesign transaction models and chain code with GolangDeploy the Composer REpresentational State Transfer (REST) Gateway to access Composer transactionsMaintain, monitor, and manage your blockchain solutionsWho this book is for This Learning Path is designed for blockchain developers who want to build decentralized applications and smart contracts from scratch using Hyperledger. Basic familiarity with or exposure to any programming language will be useful to get started with this course.

programming languages inspired by python: *Computer Dictionary* Vijay Kumar Yadav , 2024-05-14 Unlock the mysteries of computing with “Computer Dictionary.” Dive into a world of algorithms, languages, networks, and security strategies meticulously compiled to guide both novices and experts through the digital maze. Starting with the basics, Chapter 1 elucidates essential concepts like binary code, CPUs, and encryption. Move on to Chapter 2 to unravel the intricacies of programming languages, from C++ to Python. In Chapter 3, demystify networking jargon like IP addressing and VPNs, crucial for today’s interconnected landscape. Discover the backbone of computing in Chapter 4, exploring software essentials like antivirus programs and operating systems. Chapter 5 sheds light on hardware components, from RAM to GPUs, empowering readers to understand their devices better. As security concerns mount, Chapter 6 equips readers

with knowledge about biometrics, malware, and password management. Venture into Chapter 7 to explore the forefront of technology, from AI to blockchain, and Chapter 8 unveils an array of miscellaneous topics, from digital footprints to wearable tech. Concluding with a message of empowerment, “Computer Dictionary” extends its hand to students, professionals, and enthusiasts alike, offering clarity in the ever-evolving tech landscape. Embark on a journey of exploration and understanding, where technology becomes a friend, not a foe. Happy exploring!

programming languages inspired by python: *Introduction to Text Analytics* Emily Ohman, 2024-11-01 This easy-to-follow book will revolutionise how you approach text mining and data analysis as well as equipping you with the tools, and confidence, to navigate complex qualitative data. It can be challenging to effectively combine theoretical concepts with practical, real-world applications but this accessible guide provides you with a clear step-by-step approach. Written specifically for students and early career researchers this pragmatic manual will: • Contextualise your learning with real-world data and engaging case studies. • Encourage the application of your new skills with reflective questions. • Enhance your ability to be critical, and reflective, when dealing with imperfect data. Supported by practical online resources, this book is the perfect companion for those looking to gain confidence and independence whilst using transferable data skills.

programming languages inspired by python: Applied Natural Language Processing in the Enterprise Ankur A. Patel, Ajay Uppili Arasanipalai, 2021-05-12 NLP has exploded in popularity over the last few years. But while Google, Facebook, OpenAI, and others continue to release larger language models, many teams still struggle with building NLP applications that live up to the hype. This hands-on guide helps you get up to speed on the latest and most promising trends in NLP. With a basic understanding of machine learning and some Python experience, you'll learn how to build, train, and deploy models for real-world applications in your organization. Authors Ankur Patel and Ajay Uppili Arasanipalai guide you through the process using code and examples that highlight the best practices in modern NLP. Use state-of-the-art NLP models such as BERT and GPT-3 to solve NLP tasks such as named entity recognition, text classification, semantic search, and reading comprehension Train NLP models with performance comparable or superior to that of out-of-the-box systems Learn about Transformer architecture and modern tricks like transfer learning that have taken the NLP world by storm Become familiar with the tools of the trade, including spaCy, Hugging Face, and fast.ai Build core parts of the NLP pipeline--including tokenizers, embeddings, and language models--from scratch using Python and PyTorch Take your models out of Jupyter notebooks and learn how to deploy, monitor, and maintain them in production

programming languages inspired by python: *Software Languages* Talon Zinc, 2024-10-01 Code Titans: The Global Dominance of Programming Languages explores the fascinating world of programming languages that shape our digital landscape. This comprehensive guide delves into the evolution, market dominance, and real-world applications of influential languages like Python, JavaScript, and Java. The book argues that the choice of programming language significantly impacts software development efficiency and problem-solving capabilities across industries. Structured in three parts, Code Titans begins with fundamental concepts, then profiles widely-used languages, and concludes by examining future trends in programming. What sets this book apart is its holistic approach, viewing languages as living ecosystems influenced by community dynamics and global technological trends. It balances technical depth with clear explanations, making it accessible to both experienced programmers and curious non-technical readers. The book offers unique insights from interviews with language creators and industry leaders, while also exploring interdisciplinary connections between programming languages and fields like cognitive science. Readers will gain practical advice on choosing the right language for specific projects and strategies for managing multi-language software ecosystems. By understanding the strengths and limitations of today's dominant programming languages, readers will be better equipped to navigate the complex world of technology.

programming languages inspired by python: Async JavaScript Trevor Burnham, 2012-11-28

With the advent of HTML5, front-end MVC, and Node.js, JavaScript is ubiquitous--and still messy. This book will give you a solid foundation for managing async tasks without losing your sanity in a tangle of callbacks. It's a fast-paced guide to the most essential techniques for dealing with async behavior, including PubSub, evented models, and Promises. With these tricks up your sleeve, you'll be better prepared to manage the complexity of large web apps and deliver responsive code. With Async JavaScript, you'll develop a deeper understanding of the JavaScript language. You'll start with a ground-up primer on the JavaScript event model--key to avoiding many of the most common mistakes JavaScripters make. From there you'll see tools and design patterns for turning that conceptual understanding into practical code. The concepts in the book are illustrated with runnable examples drawn from both the browser and the Node.js server framework, incorporating complementary libraries including jQuery, Backbone.js, and Async.js. You'll learn how to create dynamic web pages and highly concurrent servers by mastering the art of distributing events to where they need to be handled, rather than nesting callbacks within callbacks within callbacks. Async JavaScript will get you up and running with real web development quickly. By the time you've finished the Promises chapter, you'll be parallelizing Ajax requests or running animations in sequence. By the end of the book, you'll even know how to leverage Web Workers and AMD for JavaScript applications with cutting-edge performance. Most importantly, you'll have the knowledge you need to write async code with confidence. What You Need: Basic knowledge of JavaScript is recommended. If you feel that you're not up to speed, see the Resources for Learning JavaScript section in the preface.

programming languages inspired by python: *PostgreSQL Developer's Handbook* Ewald Geschwinde, Hans-Jürgen Schönig, 2002 PostgreSQL Developer's Handbook provides a complete overview of the PostgreSQL database server and extensive coverage of its core features, including object orientation, PL/SQL, and the most important programming interfaces. The authors introduce the reader to the language and syntax of PostgreSQL and then move quickly into sophisticated programming topics.

programming languages inspired by python: Foundations of Programming Languages Kent D. Lee, 2017-12-10 This clearly written textbook provides an accessible introduction to the three programming paradigms of object-oriented/imperative, functional, and logic programming. Highly interactive in style, the text encourages learning through practice, offering test exercises for each topic covered. Review questions and programming projects are also presented, to help reinforce the concepts outside of the classroom. This updated and revised new edition features new material on the Java implementation of the JCoCo virtual machine. Topics and features: includes review questions and solved practice exercises, with supplementary code and support files available from an associated website; presents an historical perspective on the models of computation used in implementing the programming languages used today; provides the foundations for understanding how the syntax of a language is formally defined by a grammar; illustrates how programs execute at the level of assembly language, through the implementation of a stack-based Python virtual machine called JCoCo and a Python disassembler; introduces object-oriented languages through examples in Java, functional programming with Standard ML, and programming using the logic language Prolog; describes a case study involving the development of a compiler for the high level functional language Small, a robust subset of Standard ML. Undergraduate students of computer science will find this engaging textbook to be an invaluable guide to the skills and tools needed to become a better programmer. While the text assumes some background in an imperative language, and prior coverage of the basics of data structures, the hands-on approach and easy to follow writing style will enable the reader to quickly grasp the essentials of programming languages, frameworks, and architectures.

programming languages inspired by python: Programming with MicroPython Nicholas H. Tollervey, 2017-09-25 It’s an exciting time to get involved with MicroPython, the re-implementation of Python 3 for microcontrollers and embedded systems. This practical guide delivers the knowledge you need to roll up your sleeves and create exceptional embedded projects with this lean and

efficient programming language. If you're familiar with Python as a programmer, educator, or maker, you're ready to learn—and have fun along the way. Author Nicholas Tollervey takes you on a journey from first steps to advanced projects. You'll explore the types of devices that run MicroPython, and examine how the language uses and interacts with hardware to process input, connect to the outside world, communicate wirelessly, make sounds and music, and drive robotics projects. Work with MicroPython on four typical devices: PyBoard, the micro:bit, Adafruit's Circuit Playground Express, and ESP8266/ESP32 boards Explore a framework that helps you generate, evaluate, and evolve embedded projects that solve real problems Dive into practical MicroPython examples: visual feedback, input and sensing, GPIO, networking, sound and music, and robotics Learn how idiomatic MicroPython helps you express a lot with the minimum of resources Take the next step by getting involved with the Python community

programming languages inspired by python: *A Handbook of Artificial Intelligence in Drug Delivery* Anil K. Philip, Aliasgar Shahiwala, Mamoon Rashid, Md Faiyazuddin, 2023-03-27 A Handbook of Artificial Intelligence in Drug Delivery explores the use of Artificial Intelligence (AI) in drug delivery strategies. The book covers pharmaceutical AI and drug discovery challenges, Artificial Intelligence tools for drug research, AI enabled intelligent drug delivery systems and next generation novel therapeutics, broad utility of AI for designing novel micro/nanosystems for drug delivery, AI driven personalized medicine and Gene therapy, 3D Organ printing and tissue engineering, Advanced nanosystems based on AI principles (nanorobots, nanomachines), opportunities and challenges using artificial intelligence in ADME/Tox in drug development, commercialization and regulatory perspectives, ethics in AI, and more. This book will be useful to academic and industrial researchers interested in drug delivery, chemical biology, computational chemistry, medicinal chemistry and bioinformatics. The massive time and costs investments in drug research and development necessitate application of more innovative techniques and smart strategies. - Focuses on the use of Artificial Intelligence in drug delivery strategies and future impacts - Provides insights into how artificial intelligence can be effectively used for the development of advanced drug delivery systems - Written by experts in the field of advanced drug delivery systems and digital health

programming languages inspired by python: Build Your Own Programming Language Clinton L. Jeffery, 2021-12-31 Written by the creator of the Unicon programming language, this book will show you how to implement programming languages to reduce the time and cost of creating applications for new or specialized areas of computing Key Features Reduce development time and solve pain points in your application domain by building a custom programming language Learn how to create parsers, code generators, file readers, analyzers, and interpreters Create an alternative to frameworks and libraries to solve domain-specific problems Book Description The need for different types of computer languages is growing rapidly and developers prefer creating domain-specific languages for solving specific application domain problems. Building your own programming language has its advantages. It can be your antidote to the ever-increasing size and complexity of software. In this book, you'll start with implementing the frontend of a compiler for your language, including a lexical analyzer and parser. The book covers a series of traversals of syntax trees, culminating with code generation for a bytecode virtual machine. Moving ahead, you'll learn how domain-specific language features are often best represented by operators and functions that are built into the language, rather than library functions. We'll conclude with how to implement garbage collection, including reference counting and mark-and-sweep garbage collection. Throughout the book, Dr. Jeffery weaves in his experience of building the Unicon programming language to give better context to the concepts where relevant examples are provided in both Unicon and Java so that you can follow the code of your choice of either a very high-level language with advanced features, or a mainstream language. By the end of this book, you'll be able to build and deploy your own domain-specific languages, capable of compiling and running programs. What you will learn Perform requirements analysis for the new language and design language syntax and semantics Write lexical and context-free grammar rules for common expressions and control structures Develop a scanner

that reads source code and generate a parser that checks syntax Build key data structures in a compiler and use your compiler to build a syntax-coloring code editor Implement a bytecode interpreter and run bytecode generated by your compiler Write tree traversals that insert information into the syntax tree Implement garbage collection in your language Who this book is for This book is for software developers interested in the idea of inventing their own language or developing a domain-specific language. Computer science students taking compiler construction courses will also find this book highly useful as a practical guide to language implementation to supplement more theoretical textbooks. Intermediate-level knowledge and experience working with a high-level language such as Java or the C++ language are expected to help you get the most out of this book.

programming languages inspired by python: Quality of Information and Communications Technology Antonia Bertolino, João Pascoal Faria, Patricia Lago, Laura Semini, 2024-09-10 This book constitutes the proceedings of the 17th International Conference on the Quality of Information and Communications Technology, QUATIC 2024, held in Pisa, Italy, during September 11–13, 2024. The 34 full and short papers of QUATIC 2024 included in this book were carefully reviewed and selected from 49 submissions. QUATIC is a forum for disseminating advanced methods, techniques and tools to support quality approaches to ICT engineering and management. Practitioners and researchers are encouraged to exchange ideas and approaches on how to adopt a quality culture in ICT process and product improvement and to provide practical studies in varying contexts.

programming languages inspired by python: Robotics in Education Richard Balogh, David Obdržálek, Eftychios Christoforou, 2023-10-03 This book provides an overview of Educational Robotics and includes information that reflects the current status of the field, research activity, experiences, and new tools. It compiles the contributions presented at the 14th International Conference on Robotics in Education (RiE2023). Beyond insights into theoretical aspects, practical projects and syllabus activities exemplify the concepts and provide implementation ideas, which span the whole educational system from kindergarten to the university level. The relevance to science, technology, engineering, and mathematics (STEM) education is highlighted by teaching the topics in a unified framework. The book constitutes a valuable resource for educators, researchers, scientists, and engineers interested in robotics. It covers topics including school teaching curricula, educational methodologies and pedagogy, projects, competitions, hardware, simulations, programming, machine learning, and artificial intelligence in education.

programming languages inspired by python: Programming Language Pragmatics Michael Scott, 2015-11-30 Programming Language Pragmatics, Fourth Edition, is the most comprehensive programming language textbook available today. It is distinguished and acclaimed for its integrated treatment of language design and implementation, with an emphasis on the fundamental tradeoffs that continue to drive software development.The book provides readers with a solid foundation in the syntax, semantics, and pragmatics of the full range of programming languages, from traditional languages like C to the latest in functional, scripting, and object-oriented programming. This fourth edition has been heavily revised throughout, with expanded coverage of type systems and functional programming, a unified treatment of polymorphism, highlights of the newest language standards, and examples featuring the ARM and x86 64-bit architectures. - Updated coverage of the latest developments in programming language design, including C & C++11, Java 8, C# 5, Scala, Go, Swift, Python 3, and HTML 5 - Updated treatment of functional programming, with extensive coverage of OCaml - New chapters devoted to type systems and composite types - Unified and updated treatment of polymorphism in all its forms - New examples featuring the ARM and x86 64-bit architectures

Related to programming languages inspired by python

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of

transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate

your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**,

Command Line, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust

programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

What is Programming? And How to Get Started | Codecademy Programming is the mental process of thinking up instructions to give to a machine (like a computer). Coding is the process of transforming those ideas into a written language that a

Learn to Code - for Free | Codecademy Course Learn Python 3 Learn the basics of Python 3.12, one of the most powerful, versatile, and in-demand programming languages today

Learn How to Code | Codecademy New to coding? Start here and learn programming fundamentals that can be helpful for any language you learn

Code Foundations - Codecademy Start your programming journey with an introduction to the world of code and basic concepts. Includes Technical Literacy, Career Overviews, Programming Concepts, and more

What Is a Programming Language? - Codecademy Programming languages enable communication between humans and computers. Learn about how they work, the most popular languages, and their many applications

Learn the Basics of Programming with Codecademy Take this course and learn about the history and basics of programming using Blockly and pseudocode. See the specifics of different programming languages and dive into different tech

Catalog Home | Codecademy Learn the basics of the world's fastest growing and most popular

programming language used by software engineers, analysts, data scientists, and machine learning engineers alike

Java Tutorial: Learn Java Programming | Codecademy Learn to code in Java — a robust programming language used to create software, web and mobile apps, and more

Computer Science | Codecademy Looking for an introduction to the theory behind programming? Master Python while learning data structures, algorithms, and more! Includes **Python**, **Command Line**, **Git**, **Data

C++ (C Plus Plus) Courses & Tutorials | Codecademy Unlock C++ mastery with Codecademy courses & tutorials. From fundamentals to advanced concepts, enroll in our C++ courses to elevate your programming skills

Related to programming languages inspired by python

Can Python Be Used With Arduino? Here's Which Programming Languages Are Supported (SlashGear1y) Arduino is a popular brand of open-source microcontrollers that can be used for a variety of DIY projects, from educational settings to home theater to smart home management. The microcontrollers are

Can Python Be Used With Arduino? Here's Which Programming Languages Are Supported (SlashGear1y) Arduino is a popular brand of open-source microcontrollers that can be used for a variety of DIY projects, from educational settings to home theater to smart home management. The microcontrollers are

Learn popular programming languages like Python, Java, and C++ for just $70 (Mashable1y) The following content is brought to you by Mashable partners. If you buy a product featured here, we may earn an affiliate commission or other compensation. Learn the basics of Python. Credit

Learn popular programming languages like Python, Java, and C++ for just $70 (Mashable1y) The following content is brought to you by Mashable partners. If you buy a product featured here, we may earn an affiliate commission or other compensation. Learn the basics of Python. Credit

Which programming language should I use? A guide for early-career researchers (Nature5mon) Each year, the code-sharing platform GitHub releases its ‘State of the Octoverse’ report, which among other things ranks the popularity of programming languages. The latest report, released in October

Which programming language should I use? A guide for early-career researchers (Nature5mon) Each year, the code-sharing platform GitHub releases its ‘State of the Octoverse’ report, which among other things ranks the popularity of programming languages. The latest report, released in October

Want a programming job? Learn these three languages (ZDNet1y) If you want a programming job, you better ensure you're proficient in SQL, Python, and Java. That's according to the 2024 IEEE Spectrum Top Programming Languages report, which looks at what employers

Want a programming job? Learn these three languages (ZDNet1y) If you want a programming job, you better ensure you're proficient in SQL, Python, and Java. That's according to the 2024 IEEE Spectrum Top Programming Languages report, which looks at what employers

Absolute Beginner’s Guide to Python Programming: a gateway to the world of coding (Digital Trends6mon) The tech world is growing rapidly, demanding more skilled programmers. Yet, coding is still an intimidating mountain to climb for many, with its complex jargon and seemingly impenetrable logic

Absolute Beginner’s Guide to Python Programming: a gateway to the world of coding (Digital Trends6mon) The tech world is growing rapidly, demanding more skilled programmers. Yet, coding is still an intimidating mountain to climb for many, with its complex jargon and seemingly impenetrable logic

TIOBE Programming Index News May 2025: Python Hits Major Milestone (TechRepublic4mon) TIOBE Programming Index News May 2025: Python Hits Major Milestone Your email has been sent Python holds the highest share of interest in a programming language in

decades Go, Rust, and other

TIOBE Programming Index News May 2025: Python Hits Major Milestone (TechRepublic4mon) TIOBE Programming Index News May 2025: Python Hits Major Milestone Your email has been sent Python holds the highest share of interest in a programming language in decades Go, Rust, and other

The rise and fall in programming languages' popularity since 2016 - and what it tells us (ZDNet1y) Over the past few weeks, we've been discussing programming language popularity here on ZDNET. Most recently, I aggregated data from nine different rankings to produce the ZDNET Index of Programming

The rise and fall in programming languages' popularity since 2016 - and what it tells us (ZDNet1y) Over the past few weeks, we've been discussing programming language popularity here on ZDNET. Most recently, I aggregated data from nine different rankings to produce the ZDNET Index of Programming

The BASIC programming language turns 60 (Ars Technica1y) Sixty years ago, on , at 4 am in the morning, a quiet revolution in computing began at Dartmouth College. That's when mathematicians John G. Kemeny and Thomas E. Kurtz successfully ran the

The BASIC programming language turns 60 (Ars Technica1y) Sixty years ago, on , at 4 am in the morning, a quiet revolution in computing began at Dartmouth College. That's when mathematicians John G. Kemeny and Thomas E. Kurtz successfully ran the

Back to Home: https://dev.littleadventures.com