Aspectj Cookbook Wordpress

A AspectJ Developer's Notebook Specification; 150 Dotted Grid, individually numbered, cream 90g/m2 pages. Perfect matte 220g/m2 soft cover with sleek design. Customised for: AspectJ Developers and Programmers 6" x 9" dimensions; fits backpack, school, home or work. Perfect gift for adults and kids for any gift giving occasion (Christmas, Birthdays and other festive occasions.) Designed with Love by the team at 2Scribble.

This book constitutes the thoroughly refereed post-conference proceedings of the Second IFIP TC 2 Central and East-European Conference on Software Engineering Techniques, CEE-SET 2008, held in Brno, Czech Republic, in October 2008. The 20 revised full papers presented together with a keynote speech were carefully reviewed and selected from 69 initial submissions. The papers are organized in topical sections on requirements specification, design, modeling, software product lines, code generation, project management, and quality.

The AspectJ Cookbook shows readers why, and how, common Java development problems can be solved by using AOP techniques. With our popular problem-solution-discussion format, the book presents real world examples to demonstrate that AOP is more than just a concept; it's a development process that will benefit users in an immediate and visible manner.

This cookbook is written as a collection of code recipes containing step-by-step directions on how to install or build different types of Python test tools to solve different problems. Each recipe contains explanations of how it works along with answers to common questions and cross references to other relevant recipes. The easy-to-understand recipe names make this a handy test reference book.

Python developers and programmers with a basic understanding of Python and Python testing will find this cookbook beneficial. It will build on that basic knowledge equipping you with the intermediate and advanced skills required to fully utilize the Python testing tools. Broken up into lots of small code recipes, you can read this book at your own pace, whatever your experience. No prior experience of automated testing is required.

Enterprise AOP with Spring Applications

AspectJ Cookbook

Formal Approaches to Software Testing and Runtime Verification

Practical Methods for Programmer Testing

Third International Conference, DUXU 2014, Held as Part of the HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part I

150 Dotted Grid Pages Customized for AspectJ Programmers and Developers with Individually Numbered Pages. Notebook with Vibrant Colour Softcover Design. Book Format: 6 X 9 in

With forewords by Jan Bosch, Nokia and Antero Taivalsaari, Sun Microsystems. Learn how to programme the mobile devices of the future! The importance of mobile systems programming has emerged over the recent years as a new domain in software development. The design of software that runs in a mobile device requires that developers combine the rules applicable in embedded environment; memory-awareness, limited performance, security, and limited resources with features that are needed in workstation environment; modifiability, run-time extensions, and rapid application development. Programming Mobile Devices is a comprehensive, practical introduction to programming mobile systems. The book is a platform independent approach to programming mobile devices: it does not focus on specific technologies, and devices, instead it evaluates the component areas and issues that are common to all mobile software platforms. This text will enable the designer to programme mobile devices by mastering both hardware-aware and application-level software, as well as the main principles that guide their design. Programming Mobile Devices: Provides a complete and authoritative overview of programming mobile systems. Discusses the major issues surrounding mobile systems programming; such as understanding of embedded systems and workstation programming. Covers memory management, the concepts of applications, dynamically linked libraries, concurrency, handling local resources, networking and mobile devices as well as security features. Uses generic examples from JavaTM and Symbian OS to illustrate the principles of mobile device programming. Programming Mobile Devices is essential reading for graduate and advanced undergraduate students, academic and industrial researchers in the field as well as software developers, and programmers.

AspectJ shows its real power when combined with Spring. This new edition focuses on Spring-AspectJ integration, which is a major feature of Spring 2.5. Readers will find this edition immensely helpful in answering questions like: What are the ways to leverage these technologies? What applications is AOP suitable for? What are the best practices and traps? Which kind of weaving should you use? When to use Spring AOP and AspectJ AOP? Expert author Ramnivas Laddad shows how to combine technologies such as Spring, Hibernate, Swing, and JDBC with AspectJ. The book fully covers the latest AspectJ 6 features. The applications and reusable code presented in this book show how AOP vastly simplifies enterprise development. This book is for developers who have experience in AOP and AspectJ, but also for those who are new to both.

The Spring framework is growing. It has always been about choice. Java EE focused on a few technologies, largely to the detriment of alternative, better solutions. When the Spring framework debuted, few would have agreed that Java EE represented the best-in-breed architectures of the day. Spring debuted to great fanfare, because it sought to simplify Java EE. Each release since marks the introduction of new features designed to both simplify and enable solutions. With version 2.0 and later, the Spring framework started targeting multiple platforms. The framework provided services on top of existing platforms, as always, but was decoupled from the underlying platform wherever possible. Java EE is a still a major reference point, but it’s not the only target. OSGi (a promising technology for modular architectures) has been a big part of the SpringSource strategy here. Additionally, the Spring framework runs on Google App Engine. With the introduction of annotation-centric frameworks and XML schemas, SpringSource has built frameworks that effectively model the domain of a specific problem, in effect creating domain-specific languages (DSLs). Frameworks built on top of the Spring framework have emerged supporting application integration, batch processing, Flex and Flash integration, GWT, OSGi, and much more.

Globalization, security infrastructure and energy sustainability can be designed based on a scientific principle. In this book, these objectives are approached based on constructal theory, which means to design such projects as global flow architectures that are alive with movement of personnel, equipment, information, education, etc. Constructal Human Dynamics, Security and Sustainability highlights the progress made during the NATO Advanced Research Workshop held in Avora, Portugal in May 2008.

Spring Recipes

Learning UML

Approaches and Methodologies

Spring Cookbook

Mastering AspectJ

Clear, Concise, and Effective Programming

The capability modeling and simulation (M&S) supplies for managing systems complexity and investigating systems behaviors has made it a central activity in the development of new and existing systems. However, a handbook that provides established M&S practices has not been available. Until now. Modeling and Simulation-Based Systems Engineering Handbook details the M&S practices for supporting systems engineering in diverse domains. It discusses how you can identify systems engineering needs and adapt these practices to suit specific application domains, thus avoiding redefining practices from scratch. Although M&S practices are used and embedded within individual disciplines, they are often developed in isolation. However, they address recurring problems common to all disciplines. The editors of this book tackled the challenge by recruiting key representatives from several communities, harmonizing the different perspectives derived from individual backgrounds, and lining them up with the book␣s vision. The result is a collection of M&S systems engineering examples that offer an initial means for cross-domain capitalization of the knowledge, methodologies, and technologies developed in several communities. These examples provide the pros and cons of the methods and techniques available, lessons learned, and pitfalls to avoid. As our society moves further in the information era, knowledge and M&S capabilities become key enablers for the engineering of complex systems and systems of systems. Therefore, knowledge and M&S methodologies and technologies become valuable output in an engineering activity, and their cross-domain capitalization is key to further advance the future practices in systems engineering. This book collates information across disciplines to provide you with the tools to more efficiently design and manage complex systems that achieve their goals.

The four-volume set LNCS 8517, 8518, 8519 and 8520 constitutes the proceedings of the Third International Conference on Design, User Experience and Usability, DUXU 2014, held as part of the 16th International Conference on Human-Computer Interaction, HCII 2014, held in Heraklion, Crete, Greece in June 2014, jointly with 13 other thematically similar conferences. The total of 1476 papers and 220 posters presented at the HCII 2014 conferences were carefully reviewed and selected from 4766 submissions. These papers address the latest research and development efforts and highlight the human aspects of design and use of computing systems. The papers accepted for presentation thoroughly cover the entire field of Human-Computer Interaction, addressing major advances in knowledge and effective use of computers in a variety of application areas. The total of 256 contributions included in the DUXU proceedings were carefully reviewed and selected for inclusion in this four-volume set. The 66 papers included in this volume are organized in topical sections on design theories, methods and tools; user experience evaluation; heuristic evaluation; media and design; design and creativity.

Explains the concepts of aspect-oriented programming and the basics of the AspectJ language.

This journal is devoted to aspect-oriented software development (AOSD) techniques in the context of all phases of the software life cycle, from requirements and design to implementation, maintenance and evolution. The focus of the journal is on approaches for systematic identification, modularization, representation and composition of crosscutting concerns, evaluation of such approaches and their impact on improving quality attributes of software systems.

Constructal Human Dynamics, Security and Sustainability

Software Engineering Techniques

Design, User Experience, and Usability: Theories, Methods, and Tools for Designing the User Experience

"Real-world Aspect-oriented Programming with Java"--Cover. - Includes Index

Eclipse Aspectj

AspectJ in Action

Pro Spring 2 is the perfect, simple answer for your lightweight, alternative Java EE development needs! Put simply, this book brings J2EE/Java EE "down to earth." Without the hassles of using the EJB 3 specification and similar, you can build lighter, better-performing agile enterprise Java-based applications using Spring Framework 2. The Spring framework can also integrate other noteworthy and hot open source tools like Apache Struts, Hibernate, OpenJPA, GlassFish, and many more. You’ll work through a real, scalable enterprise application and build it from the ground up with Spring, using all the multiple web views and frameworks.

Svenja Hager aims at pricing non-standard illiquid portfolio credit derivatives which are related to standard CDO tranches with the same underlying portfolio of obligors. Instead of assuming a homogeneous dependence structure between the default times of different obligors, as it is assumed in the standard market model, the author focuses on the use of heterogeneous correlation structures.

Gives experienced Java developers the tools to exploit aspect-oriented programming techniques using AspectJ, an open source Java extension Delivers a code-intensive, real-world tutorial on building applications with AspectJ Covers the AspectJ compiler and browser as well as the IDE plug-ins and other tools that can be used with AspectJ Masterfully ties together all material in the book so that readers will be able to build a complete, working application Companion Web site includes all sample code, the complete application, and links to other relevant sites

Centered on the impact of information and communication technology in socio-technical environments and its support of human activity systems, the study of information systems remains a distinctive focus in the area of computer science research. Information Systems Research and Exploring Social Artifacts: Approaches and Methodologies discusses the approaches and methodologies currently being used in the field on information systems. This reference source covers a wide variety of socio-technical aspects of the design of IS artifacts as well as the study of their use. This book aims to be useful for researchers, scholars and students interested in expanding their knowledge on the assortment of research on information systems.

An Introduction for Practitioners

Aspectj Cookbook

Modeling and Simulation-Based Systems Engineering Handbook

Next Level Software Engineering with Data Warehouses

Programming Mobile Devices

Fluent Python

This book constitutes the thoroughly refereed post-proceedings of the First Combined International Workshops on Formal Approaches to Software Testing, FATES 2006, and on Runtime Verification, RV 2006, held within the scope of FLoC 2006, the Federated Logic Conference in Seattle, WA, USA in August 2006. Coverage discusses formal approaches to test and analyze programs and monitor and guide their executions by using various techniques.

Learn how to successfully implement trustworthy computing tasks using aspect-oriented programming This landmark publication fills a gap in the literature by not only describing the basic concepts of trustworthy computing (TWC) and aspect-oriented programming (AOP), but also exploring their critical interrelationships. The author clearly demonstrates how typical TWC tasks such as security checks, in-and-out conditions, and multi-threaded safety can be implemented using AOP. Following an introduction, the book covers: Trustworthy computing, software engineering, and computer science Aspect-oriented programming and Aspect.NET Principles and case studies that apply AOP to TWC Coverage includes Aspect.NET, the AOP framework developed by the author for the Microsoft.NET platform, currently used in seventeen countries. The author discusses the basics of Aspect.NET architecture, its advantages compared to other AOP tools, and its functionality. The book has extensive practical examples and case studies of trustworthy software design and code using the Aspect.NET framework. In addition, the book explores other software technologies and tools for using AOP for trustworthy software development, including Java and AspectJ. This book also includes a valuable chapter dedicated to ERATO, the author's teaching method employed in this book, which has enabled thousands of students to quickly grasp and apply complex concepts in computing and software engineering, while the final chapter presents an overall perspective on the current state of AOP and TWC with a view toward the future. Software engineers, architects, developers, programmers, and students should all turn to this book to learn this tested and proven method to create more secure, private, and reliable computing.

This hands-on book shows readers why and how common Java development problems can be solved by using new Aspect-oriented programming (AOP) techniques. With a wide variety of code recipes for solving day-to-day design and coding problems using AOP's unique approach, 'AspectJ Cookbook' demonstrates that AOP is more than just a concept.

To allow the creation of truly modular software, OOP has evolved into aspect-oriented programming. AspectJ is a mature AOP implementation for Java, now integrated with Spring. AspectJ in Action, Second Edition is a fully updated, major revision of Ramnivas Laddad's best-selling first edition. It's a hands-on guide for Java developers. After introducing the core principles of AOP, it shows you how to create reusable solutions using AspectJ 6 and Spring 3. You'll master key features including annotation-based syntax, load-time weaver, annotation-based crosscutting, and Spring-AspectJ integration. Building on familiar technologies such as JDBC, Hibernate, JPA, Spring Security, Spring MVC, and Swing, you'll apply AOP to common problems encountered in enterprise applications. This book requires no previous experience in AOP and AspectJ, but it assumes you're familiar with OOP, Java, and the basics of Spring. "Clear, concisely worded, well-organized ... a pleasure to read." -From the Foreword by Rod Johnson, Creator of the Spring Framework "This book teaches you how to think in aspects. It is essential reading for both beginners who know nothing about AOP and experts who think they know it all." - Andrew Eisenberg, AspectJ Development Tools Project Committer "Ramnivas showcases how to get the best out of AspectJ and Spring." -Andy Clement, AspectJ Project Lead "One of the best Java books in years." -Andrew Rhine, Software Engineer, eSecLending "By far the best reference for Spring AOP and AspectJ." -Paul Benedict, Software Engineer, Argus Health Systems "Ramnivas expertly demystifies the awesome power of aspect-oriented programming." -Craig Walls, author of Spring in Action

Pricing Portfolio Credit Derivatives by Means of Evolutionary Algorithms

Spring 5 Recipes

Third IFIP TC 2 Central and East-European Conference, CEE-SET 2008, Brno, Czech Republic, October 13-15, 2008, Revised Selected Papers

Spring Roo 1.1 Cookbook

Hypermodelling

Aspect-oriented Programming with Aspectj and the Eclipse Aspectj Development Tools

The book constitutes the refereed proceedings of the 10th International Conference on Software Composition, SC 2011, held in Zurich, Switzerland, in June/July 2011, co-located with TOOLS 2011 Federated Conferences. The 10 revised full papers and 2 short papers were carefully reviewed and selected from 32 initial submissions for inclusion in the book. The papers reflect all current research in software composition and are organized in topical sections on composition and interfaces, aspects and features, and applications.

???cookbook?????"TM"

Advanced Techniques in Computing Sciences and Software Engineering includes a set of rigorously reviewed world-class manuscripts addressing and detailing state-of-the-art research projects in the areas of Computer Science, Software Engineering, Computer Engineering, and Systems Engineering and Sciences. Advanced Techniques in Computing Sciences and Software Engineering includes selected papers form the conference proceedings of the International Conference on Systems, Computing Sciences and Software Engineering (SCSS 2008) which was part of the International Joint Conferences on Computer, Information and Systems Sciences and Engineering (CISSE 2008).

AspectJ Cookbook"O'Reilly Media, Inc."

Python in a Nutshell

JUnit Recipes

A AspectJ Developer's Notebook

On the separation of user interface concerns: A Programmer's Perspective on the Modularisation of User Interface Code

Software Composition

A Problem-Solution Approach

Solve all your Spring 5 problems using complete and real-world code examples. When you start a new project, you’ll be able to copy the code and configuration files from this book, and then modify them for your needs. This can save you a great deal of work over creating a project from scratch. The recipes in Spring 5 Recipes cover Spring fundamentals such as Spring IoC container, Spring AOP/ AspectJ, and more. Other recipes include Spring enterprise solutions for topics such as Spring Java EE integration, Spring Integration, Spring Batch, Spring Remoting, messaging, transactions, and working with big data and the cloud using Hadoop and MongoDB. Finally, Spring web recipes cover Spring MVC, other dynamic scripting, integration with the popular Grails Framework (and Groovy), REST/web services, and more. You’ll also see recipes on new topics such as Spring Framework 5, reactive Spring, Spring 5 microservices, the functional web framework and much more. This book builds upon the best-selling success of the previous editions and focuses on the latest Spring Framework features for building enterprise Java applications. What You'll Learn Get re-usable code recipes and snippets for core Spring, annotations and other development tools Access Spring MVC for web development Work with Spring REST and microservices for web services development and integration into your enterprise Java applications Use Spring Batch, NoSQL and big data for building and integrating various cloud computing services and resources Integrate Java Enterprise Edition and other Java APIs for use in Spring Use Grails code and much more Who This Book Is For Experienced Java and Spring programmers.

This new book is the definitive primer for UML, and starts with the foundational concepts of object-orientation in order to provide the proper context for explaining UML.

This book is for you if you have some experience with Java and web development (not necessarily in Java) and want to become proficient quickly with Spring.

Over 60 recipes to help you speed up the development of your Java web applications using the Spring Roo development tool.

First Combined International Workshops FATES 2006 and RV 2006, Seattle, WA, USA, August 15-16, 2006, Revised Selected Papers

Practical Aspect-oriented Programming

Python Testing Cookbook

Transactions on Aspect-Oriented Software Development I

AspectJ Cookbook[TM]□□□

Aspect-Oriented Programming in Java

Optimized for Kubernetes, Quarkus is designed to help you create Java applications that are cloud first, container native, and serverless capable. With this cookbook, authors Alex Soto Bueno and Jason Porter from Red Hat provide detailed solutions for installing, interacting with, and using Quarkus in the development and production of microservices. The recipes in this book show midlevel to senior developers familiar with Java enterprise application development how to get started with Quarkus quickly. You'll become familiar with how Quarkus works within the wider Java ecosystem and discover ways to adapt this framework to your particular needs. You'll learn how to: Shorten the development cycle by enabling live reloading in dev mode Connect to and communicate with Kafka Develop with the reactive programming model Easily add fault tolerance to your services Build your application as a Kubernetes-ready container Ease development with OpenAPI and test a native Quarkus application

Do you want to push Ruby to its limits? The Ruby Cookbook is the most comprehensive problem-solving guide to today's hottest programming language. It gives you hundreds of solutions to real-world problems, with clear explanations and thousands of lines of code you can use in your own projects. From data structures and algorithms, to integration with cutting-edge technologies, the Ruby Cookbook has something for every programmer. Beginners and advanced Rubyists alike will learn how to program with: Strings and numbers Arrays and hashes Classes, modules, and namespaces Reflection and metaprogramming XML and HTML processing Ruby on Rails (including Ajax integration) Databases Graphics Internet services like email, SSH, and BitTorrent Web services Multitasking Graphical and terminal interfaces If you need to write a web application, this book shows you how to get started with Rails. If you're a system administrator who needs to rename thousands of files, you'll see how to use Ruby for this and other everyday tasks. You'll learn how to read and write Excel spreadsheets, classify text with Bayesian filters, and create PDF files. We've even included a few silly tricks that were too cool to leave out, like how to blink the lights on your keyboard. The Ruby Cookbook is the most useful book yet written about Ruby. When you need to solve a problem, don't reinvent the wheel: look it up in the Cookbook.

"Mastering AspectJ" shows experienced Java developers how to exploit aspect-oriented programming techniques using AspectJ. It begins by providing an overview of the concepts of AOP and of the AspectJ language, then moves quickly to provide a code-intensive, real-world tutorial on building applications.

The definitive (and only) introduction to Aspect-Oriented Programming (AOP) using Eclipse and ASpectJ.

Advanced Techniques in Computing Sciences and Software Engineering

Information Systems Research and Exploring Social Artifacts: Approaches and Methodologies

Using Aspect-Oriented Programming for Trustworthy Software Development

Aspect-oriented Software Development with Use Cases

Aspect-oriented Programming with AspectJ

When testing becomes a developer's habit good things tend to happen--good productivity, good code, and good job satisfaction. If you want some of that, there's no better way to start your testing habit, nor to continue feeding it, than with"" JUnit Recipes,"" In this book you will find one hundred and thirty-seven solutions to a range of problems, from simple to complex, selected for you by an experienced developer and master tester. Each recipe follows the same organization giving you the problem and its background before discussing your options in solving it. JUnit - the unit testing framework for Java - is simple to use, but some code can be tricky to test. When you're facing such code you will be glad to have this book. It is a how-to reference full of practical advice on all issues of testing, from how to name your test case classes to how to test complicated J2EE applications. Its valuable advice includes side matters that can have a big payoff, like how to organize your test data or how to manage expensive test resources. What's Inside: - Getting started with JUnit - Recipes for: servlets JSPs EJBs Database code much more - Difficult-to-test designs, and how to fix them - How testing saves time - Choose a JUnit extension: HTMLUnit XMLUnit ServletUnit EasyMock and more!

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.

"A refreshingly new approach toward improving use-case modeling by fortifying it with aspect orientation." --Ramnivas Laddad, author of AspectJ in Action "Since the 1980s, use cases have been a way to bring users into software design, but translating use cases into software has been an art, at best, because user goods often don''t respect code boundaries. Now that aspect-oriented programming (AOP) can express crosscutting concerns directly in code, the man who developed use cases has proposed step-by-step methods for recognizing crosscutting concerns in use cases and writing the code in separate modules. If these methods are at all fruitful in your design and development practice, they will make a big difference in software quality for developers and users alike. --Wes Isberg, AspectJ team member"This book not only provides ideas and examples of what aspect-oriented software development is but how it can be utilized in a real development project." --MichaelWard, ThoughtWorks, Inc."No system has ever been designed from scratch perfectly; every system is composed of features layered in top of features that accumulate over time. Conventional design techniques do not handle this well, and over time the integrity of most systems degrades as a result. For the first time, here is a set of techniques that facilitates composition of behavior that not only allows systems to be defined in terms of layered functionality but composition is at the very heart of the approach. This book is an important advance in modern methodology and is certain to influence the direction of software engineering in the next decade, just as Object-Oriented Software Engineering influenced the last." --Kurt Bittner, IBM Corporation"Use cases are an excellent means to capture system requirements and drive a user-centric view of system development and testing. This book offers a comprehensive guide on explicit use-case-driven development from early requirements modeling to design and implementation. It provides a simple yet rich set of guidelines to realize use-case models using aspect-oriented design and programming. It is a valuable resource to researchers and practitioners alike." --Dr. Awais Rashid, Lancaster University, U.K., and author of Aspect-Oriented Database Systems "AOSD is important technology that will help developers produce better systems. Unfortunately, it has not been obvious how to integrate AOSD across a project''s lifecycle. This book shatters that barrier, providing concrete examples on how to use AOSD from requirements analysis through testing." --Charles B. Haley, research fellow, The Open University, U.K. Aspect-oriented programming (AOP) is a revolutionary new way to think about software engineering. AOP was introduced to address crosscutting concerns such as security, logging, persistence, debugging, tracing, distribution, performance monitoring, and exception handling in a more effective manner. Unlike conventional development techniques, which scatter the implementation of each concern into multiple classes, aspect-oriented programming localizes them. Aspect-oriented software development (AOSD) uses this approach to create a better modularity for functional and nonfunctional requirements, platform specifics, and more, allowing you to build more understandable systems that are easier to configure and extend to meet the evolving needs of stakeholders. In this highly anticipated new book, Ivar Jacobson and Pan-Wei Ng demonstrate how to apply use cases--a mature and systematic approach to focusing on stakeholder concerns--and aspect-orientation in building robust and extensible systems. Throughout the book, the authors employ a single, real-world example of a hotel management information system to make the described theories and practices concrete and understandable. The authors show how to identify, design, implement, test, and refactor use-case modules, as well as extend them. They also demonstrate how to design use-case modules with the Unified Modeling Language (UML)--emphasizing enhancements made in UML 2.0--and how to achieve use-case modularity using aspect technologies, notably AspectJ. Key topics include Making the case for use cases and aspects Capturing and modeling concerns with use cases Keeping concerns separate with use-case modules Modeling use-cases slices and aspects using the newest extensions to the UML notation Applying use cases and aspects in projects Whatever your level of experience with aspect-oriented programming, Aspect-Oriented Software Development with Use Cases will teach you how to develop better software by embracing the paradigm shift to AOSD.

In a November 2001 Java Pro magazine article, noted Java pundit Daniel Savarese states, "The days of Object-Oriented Programming may be numbered, One day we may all be using Aspect-Oriented Programming ." While this may be hyperbole, the AOP bring certain needed improvements to the OOP. AspectJ is a Java-based tool that allows developers to apply standard Java syntax to AOP principles, much as C++ allowed C programmers to use C syntax in an object-oriented manner. There are AspectJ add-ons available for Borland's JBuilder, Sun's Forte for Java and for the EMACS text editor. Aspect-Oriented Programming with AspectJ introduces AOP and the AspectJ tool. The book also shows how, by using existing Java programming knowledge, the developer can use AOP in meaningful development work.

Quarkus Cookbook

10th International Conference, SC 2011, Zurich, Switzerland, June 30 - July 1, 2011, Proceedings

Pro Spring 2.5

Transactions on Aspect-Oriented Software Development VIII

Ruby Cookbook

Python’s simplicity lets you become productive quickly, but this often means you aren’t using everything it has to offer. With this hands-on guide, you’ll learn how to write effective, idiomatic Python code by leveraging its best—and possibly most neglected—features. Author Luciano Ramalho takes you through Python’s core language features and libraries, and shows you how to make your code shorter, faster, and more readable at the same time. Many experienced programmers try to bend Python to fit patterns they learned from other languages, and never discover Python features outside of their experience. With this book, those Python programmers will thoroughly learn how to become proficient in Python 3. This book covers: Python data model: understand how special methods are the key to the consistent behavior of objects Data structures: take full advantage of built-in types, and understand the text vs bytes duality in the Unicode age Functions as objects: view Python functions as first-class objects, and understand how this affects popular design patterns Object-oriented idioms: build classes by learning about references, mutability, interfaces, operator overloading, and multiple inheritance Control flow: leverage context managers, generators, coroutines, and concurrency with the concurrent.futures and asyncio packages Metaprogramming: understand how properties, attribute descriptors, class decorators, and metaclasses work

This volume, the 8th in the Transactions on Aspect-Oriented Software Development series, contains two regular submissions and a special section, consisting of five papers, on the industrial applications of aspect technology. The regular papers describe a framework for constructing aspect weavers, and patterns for reusable aspects. The special section begins with an invited contribution on how AspectJ is making its way from an exciting new hype topic to a valuable technology in enterprise computing. The remaining four papers each cover different industrial applications of aspect technology, which include a telecommunication platform, a framework for embedding user assistance in independently developed applications, a platform for digital publishing, and a framework for program code analysis and manipulation.