

Refactoring Databases Evolutionary Database Design

Right here, we have countless books **Refactoring Databases Evolutionary Database Design** and collections to check out. We additionally manage to pay for variant types and with type of the books to browse. The standard book, fiction, history, novel, scientific research, as with ease as various further sorts of books are readily straightforward here.

As this Refactoring Databases Evolutionary Database Design , it ends occurring living thing one of the favored books Refactoring Databases Evolutionary Database Design collections that we have. This is why you remain in the best website to see the amazing ebook to have.

SQL Queries for Mere Mortals - John L. Viescas 2014

Presents a guide to writing effective SQL queries, from simple data selection and filtering to joining multiple tables and modifying sets of data, with information on how to solve a variety of challenging SQL problems.

Choose Your WoW! - Scott W. Ambler 2020

"Hundreds of organizations around the world have already benefited from Disciplined Agile Delivery (DAD). Disciplined Agile (DA) is the only comprehensive tool kit available for guidance on building high-performance agile teams and optimizing your way of working (WoW). As a hybrid of all the leading agile and lean approaches, it provides hundreds of strategies to help you make better decisions within your agile teams, balancing self-organization with the realities and constraints of your unique enterprise context. The highlights of this handbook include: #1. As the official source of knowledge on DAD, it includes greatly improved and enhanced strategies with a revised set of goal diagrams based upon learnings from applying DAD in the field. #2 It is an essential handbook to help coaches and teams make better decisions in their daily work, providing a wealth of ideas for experimenting with agile and lean techniques while providing specific guidance and trade-offs for those "it depends" questions. #3 It makes a perfect study guide for Disciplined Agile certification. Why "fail fast" (as our industry likes to recommend) when you can learn quickly on your journey to high performance? With this handbook, you can make better decisions based upon proven, context-based strategies, leading to earlier success and better outcomes"--

Refactoring - Paul Becker 1999

Refactoring is gaining momentum amongst the object oriented programming community. It can transform the internal dynamics of applications and has the capacity to transform bad code into good code. This book offers an introduction to refactoring.

The Elements of Java(TM) Style - Allan Vermeulen 2000-01-28

The Elements of Java Style, written by renowned author Scott Ambler, Rogue Wave Software Vice President Alan Vermeulen, and a team of programmers from Rogue Wave, is for anyone who writes Java code. While there are many books that explain the syntax and basic use of Java, this book, first published in 2000, explains not just what you can do with the syntax, but what you ought to do. Just as Strunk and White's The Elements of Style provides rules of usage for the English language, this book provides a set of rules for Java practitioners to follow. While illustrating these rules with parallel examples of correct and incorrect usage, the book provides a collection of standards, conventions, and guidelines for writing solid Java code which will be easy to understand, maintain, and enhance. Anyone who writes Java code or plans to should have this book next to their computer.

Refactoring Databases - Ambler 2006

Agile Database Techniques - Scott Ambler 2012-09-17

Describes Agile Modeling Driven Design (AMDD) and Test-Driven Design (TDD) approaches, database refactoring, database encapsulation strategies, and tools that support evolutionary techniques Agile software developers often use object and relational database (RDB) technology together and as a result must overcome the impedance mismatch The author covers techniques for mapping objects to RDBs and for implementing concurrency control, referential integrity, shared business logic, security access control,

reports, and XML An agile foundation describes fundamental skills that all agile software developers require, particularly Agile DBAs Includes object modeling, UML data modeling, data normalization, class normalization, and how to deal with legacy databases Scott W. Ambler is author of Agile Modeling (0471202827), a contributing editor with Software Development (www.sdmagazine.com), and a featured speaker at software conferences worldwide

Balancing Agility and Formalism in Software Engineering - Bertrand Meyer 2008-07-31

This book constitutes the thoroughly refereed post-conference proceedings of the Second IFIP TC 2 Central and East Conference on Software Engineering Techniques, CEE-SET 2007, held in Poznan, Poland, in October 2007. The 21 revised full papers presented together with 2 keynote addresses were carefully reviewed and selected from 73 initial submissions. The papers are organized in topical sections on measurement, processes, UML, experiments, tools, and change.

Software Architecture: The Hard Parts - Neal Ford 2021-09-23

There are no easy decisions in software architecture. Instead, there are many hard parts--difficult problems or issues with no best practices--that force you to choose among various compromises. With this book, you'll learn how to think critically about the trade-offs involved with distributed architectures. Architecture veterans and practicing consultants Neal Ford, Mark Richards, Pramod Sadalage, and Zhamak Dehghani discuss strategies for choosing an appropriate architecture. By interweaving a story about a fictional group of technology professionals--the Sysops Squad--they examine everything from how to determine service granularity, manage workflows and orchestration, manage and decouple contracts, and manage distributed transactions to how to optimize operational characteristics, such as scalability, elasticity, and performance. By focusing on commonly asked questions, this book provides techniques to help you discover and weigh the trade-offs as you confront the issues you face as an architect. Analyze trade-offs and effectively document your decisions Make better decisions regarding service granularity Understand the complexities of breaking apart monolithic applications Manage and decouple contracts between services Handle data in a highly distributed architecture Learn patterns to manage workflow and transactions when breaking apart applications

Practical Applications of Data Mining - Sang Suh 2012

Introduction to data mining -- Association rules -- Classification learning -- Statistics for data mining -- Rough sets and bayes theories -- Neural networks -- Clustering -- Fuzzy information retrieval.

The Object Primer - Scott W. Ambler 2001-05-28

Scott Ambler, author of Building Object Applications that Work, Process Patterns, and More Process Patterns, has revised his acclaimed first book, The Object Primer. Long prized in its original edition by both students and professionals as the best introduction to object-oriented technology, now this book is completely up-to-date with new material in every chapter. There are also new chapters on good OO programming techniques and OO software testing. All modeling notation has been rewritten in UML notation. Review questions at the end of each chapter allow readers to test their newly acquired knowledge. In addition, the author takes time to reflect on the lessons learned over the past few years by discussing the proven benefits and drawbacks of the technology. This is the perfect book for any software development professional or student seeking an introduction to the concepts and terminology of object technology.

Agile Analytics - Ken Collier 2012

Using Agile methods, you can bring far greater innovation, value, and quality to any data warehousing (DW), business intelligence (BI), or analytics project. However, conventional Agile methods must be carefully adapted to address the unique characteristics of DW/BI projects. In Agile Analytics, Agile pioneer Ken Collier shows how to do just that. Collier introduces platform-agnostic Agile solutions for integrating infrastructures consisting of diverse operational, legacy, and specialty systems that mix commercial and custom code. Using working examples, he shows how to manage analytics development teams with widely diverse skill sets and how to support enormous and fast-growing data volumes. Collier's techniques offer optimal value whether your projects involve "back-end" data management, "front-end" business analysis, or both. Part I focuses on Agile project management techniques and delivery team coordination, introducing core practices that shape the way your Agile DW/BI project community can collaborate toward success Part II presents technical methods for enabling continuous delivery of business value at production-quality levels, including evolving superior designs; test-driven DW development; version control; and project automation Collier brings together proven solutions you can apply right now--whether you're an IT decision-maker, data warehouse professional, database administrator, business intelligence specialist, or database developer. With his help, you can mitigate project risk, improve business alignment, achieve better results--and have fun along the way.

Evolve the Monolith to Microservices with Java and Node - Sandro De Santis 2016-12-05

Microservices is an architectural style in which large, complex software applications are composed of one or more smaller services. Each of these microservices focuses on completing one task that represents a small business capability. These microservices can be developed in any programming language. This IBM® Redbooks® publication shows how to break out a traditional Java EE application into separate microservices and provides a set of code projects that illustrate the various steps along the way. These code projects use the IBM WebSphere® Application Server Liberty, IBM API Connect™, IBM Bluemix®, and other Open Source Frameworks in the microservices ecosystem. The sample projects highlight the evolution of monoliths to microservices with Java and Node.

Fowler - Martin Fowler 2012-03-09

The practice of enterprise application development has benefited from the emergence of many new enabling technologies. Multi-tiered object-oriented platforms, such as Java and .NET, have become commonplace. These new tools and technologies are capable of building powerful applications, but they are not easily implemented. Common failures in enterprise applications often occur because their developers do not understand the architectural lessons that experienced object developers have learned. Patterns of Enterprise Application Architecture is written in direct response to the stiff challenges that face enterprise application developers. The author, noted object-oriented designer Martin Fowler, noticed that despite changes in technology--from Smalltalk to CORBA to Java to .NET--the same basic design ideas can be adapted and applied to solve common problems. With the help of an expert group of contributors, Martin distills over forty recurring solutions into patterns. The result is an indispensable handbook of solutions that are applicable to any enterprise application platform. This book is actually two books in one. The first section is a short tutorial on developing enterprise applications, which you can read from start to finish to understand the scope of the book's lessons. The next section, the bulk of the book, is a detailed reference to the patterns themselves. Each pattern provides usage and implementation information, as well as detailed code examples in Java or C#. The entire book is also richly illustrated with UML diagrams to further explain the concepts. Armed with this book, you will have the knowledge necessary to make important architectural decisions about building an enterprise application and the proven patterns for use when building them. The topics covered include · Dividing an enterprise application into layers · The major approaches to organizing business logic · An in-depth treatment of mapping between objects and relational databases · Using Model-View-Controller to organize a Web presentation · Handling concurrency for data that spans multiple transactions · Designing distributed object interfaces

The Art of Agile Development - James Shore 2008

For those considering Extreme Programming, this book provides no-nonsense advice on agile planning, development, delivery, and management taken from the authors' many years of experience. While plenty of books address the what and why of agile development, very few offer the information users can apply

directly.

Service Design Patterns - Robert Daigneau 2011-10-25

Web services have been used for many years. In this time, developers and architects have encountered a number of recurring design challenges related to their usage, and have learned that certain service design approaches work better than others to solve certain problems. In Service Design Patterns, Rob Daigneau codifies proven design solutions for web services that follow the REST architectural style or leverage the SOAP/WSDL specifications. This catalogue identifies the fundamental topics in web service design and lists the common design patterns for each topic. All patterns identify the context in which they may be used, explain the constituent design elements, and explore the relative strengths and trade-offs. Code examples are provided to help you better understand how the patterns work but are kept general so that you can see how the solutions may be applied to disparate technologies that will inevitably change in the years to come. This book will help readers answer the following questions: How do you create a web service API, what are the common API styles, and when should a particular style be used? How can clients and web services communicate, and what are the foundations for creating complex conversations in which multiple parties exchange data over extended periods of time? What are the options for implementing web service logic, and when should a particular approach be used? How can clients become less coupled to the underlying systems used by a service? How can information about a web service be discovered? How can generic functions like authentication, validation, caching, and logging be supported on the client or service? What changes to a service cause clients to break? What are the common ways to version a service? How can web services be designed to support the continuing evolution of business logic without forcing clients to constantly upgrade? This book is an invaluable resource for enterprise architects, solution architects, and developers who use web services to create enterprise IT applications, commercial or open source products, and Software as a Service (SaaS) products that leverage emerging Cloud platforms.

Refactoring for Software Design Smells - Girish Suryanarayana 2014-11-11

Awareness of design smells - indicators of common design problems - helps developers or software engineers understand mistakes made while designing, what design principles were overlooked or misapplied, and what principles need to be applied properly to address those smells through refactoring. Developers and software engineers may "know" principles and patterns, but are not aware of the "smells" that exist in their design because of wrong or mis-application of principles or patterns. These smells tend to contribute heavily to technical debt - further time owed to fix projects thought to be complete - and need to be addressed via proper refactoring. Refactoring for Software Design Smells presents 25 structural design smells, their role in identifying design issues, and potential refactoring solutions. Organized across common areas of software design, each smell is presented with diagrams and examples illustrating the poor design practices and the problems that result, creating a catalog of nuggets of readily usable information that developers or engineers can apply in their projects. The authors distill their research and experience as consultants and trainers, providing insights that have been used to improve refactoring and reduce the time and costs of managing software projects. Along the way they recount anecdotes from actual projects on which the relevant smell helped address a design issue. Contains a comprehensive catalog of 25 structural design smells (organized around four fundamental design principles) that contribute to technical debt in software projects Presents a unique naming scheme for smells that helps understand the cause of a smell as well as points toward its potential refactoring Includes illustrative examples that showcase the poor design practices underlying a smell and the problems that result Covers pragmatic techniques for refactoring design smells to manage technical debt and to create and maintain high-quality software in practice Presents insightful anecdotes and case studies drawn from the trenches of real-world projects

The Data Model Resource Book, Volume 1 - Len Silverston 2011-08-08

A quick and reliable way to build proven databases for core business functions Industry experts raved about The Data Model Resource Book when it was first published in March 1997 because it provided a simple, cost-effective way to design databases for core business functions. Len Silverston has now revised and updated the hugely successful 1st Edition, while adding a companion volume to take care of more specific requirements of different businesses. This updated volume provides a common set of data models for specific core functions shared by most businesses like human resources management, accounting, and

project management. These models are standardized and are easily replicated by developers looking for ways to make corporate database development more efficient and cost effective. This guide is the perfect complement to The Data Model Resource CD-ROM, which is sold separately and provides the powerful design templates discussed in the book in a ready-to-use electronic format. A free demonstration CD-ROM is available with each copy of the print book to allow you to try before you buy the full CD-ROM.

Database Design for Mere Mortals - Michael James Hernandez 2003

"This book takes the somewhat daunting process of database design and breaks it into completely manageable and understandable components. Mike's approach whilst simple is completely professional, and I can recommend this book to any novice database designer." --Sandra Barker, Lecturer, University of South Australia, Australia "Databases are a critical infrastructure technology for information systems and today's business. Mike Hernandez has written a literate explanation of database technology--a topic that is intricate and often obscure. If you design databases yourself, this book will educate you about pitfalls and show you what to do. If you purchase products that use a database, the book explains the technology so that you can understand what the vendor is doing and assess their products better." --Michael Blaha, consultant and trainer, author of A Manager's Guide to Database Technology "If you told me that Mike Hernandez could improve on the first edition of Database Design for Mere Mortals I wouldn't have believed you, but he did! The second edition is packed with more real-world examples, detailed explanations, and even includes database-design tools on the CD-ROM! This is a must-read for anyone who is even remotely interested in relational database design, from the individual who is called upon occasionally to create a useful tool at work, to the seasoned professional who wants to brush up on the fundamentals. Simply put, if you want to do it right, read this book!" --Matt Greer, Process Control Development, The Dow Chemical Company "Mike's approach to database design is totally common-sense based, yet he's adhered to all the rules of good relational database design. I use Mike's books in my starter database-design class, and I recommend his books to anyone who's interested in learning how to design databases or how to write SQL queries." --Michelle Poolet, President, MVDS, Inc. "Slapping together sophisticated applications with poorly designed data will hurt you just as much now as when Mike wrote his first edition, perhaps even more. Whether you're just getting started developing with data or are a seasoned pro; whether you've read Mike's previous book or this is your first; whether you're happier letting someone else design your data or you love doing it yourself--this is the book for you. Mike's ability to explain these concepts in a way that's not only clear, but fun, continues to amaze me." --From the Foreword by Ken Getz, MCW Technologies, coauthor ASP.NET Developer's JumpStart "The first edition of Mike Hernandez's book Database Design for Mere Mortals was one of the few books that survived the cut when I moved my office to smaller quarters. The second edition expands and improves on the original in so many ways. It is not only a good, clear read, but contains a remarkable quantity of clear, concise thinking on a very complex subject. It's a must for anyone interested in the subject of database design." --Malcolm C. Rubel, Performance Dynamics Associates "Mike's excellent guide to relational database design deserves a second edition. His book is an essential tool for fledgling Microsoft Access and other desktop database developers, as well as for client/server pros. I recommend it highly to all my readers." --Roger Jennings, author of Special Edition Using Access 2002 "There are no silver bullets! Database technology has advanced dramatically, the newest crop of database servers perform operations faster than anyone could have imagined six years ago, but none of these technological advances will help fix a bad database design, or capture data that you forgot to include! Database Design for Mere Mortals(TM), Second Edition, helps you design your database right in the first place!" --Matt Nunn, Product Manager, SQL Server, Microsoft Corporation "When my brother started his professional career as a developer, I gave him Mike's book to help him understand database concepts and make real-world application of database technology. When I need a refresher on the finer points of database design, this is the book I pick up. I do not think that there is a better testimony to the value of a book than that it gets used. For this reason I have wholeheartedly recommended to my peers and students that they utilize this book in their day-to-day development tasks." --Chris Kunicki, Senior Consultant, OfficeZealot.com "Mike has always had an incredible knack for taking the most complex topics, breaking them down, and explaining them so that anyone can 'get it.' He has honed and polished his first very, very good edition and made it even better. If you're just starting out building database applications, this book is

a must-read cover to cover. Expert designers will find Mike's approach fresh and enlightening and a source of great material for training others." --John Viescas, President, Viescas Consulting, Inc., author of Running Microsoft Access 2000 and coauthor of SQL Queries for Mere Mortals "Whether you need to learn about relational database design in general, design a relational database, understand relational database terminology, or learn best practices for implementing a relational database, Database Design for Mere Mortals(TM), Second Edition, is an indispensable book that you'll refer to often. With his many years of real-world experience designing relational databases, Michael shows you how to analyze and improve existing databases, implement keys, define table relationships and business rules, and create data views, resulting in data integrity, uniform access to data, and reduced data-entry errors." --Paul Cornell, Site Editor, MSDN Office Developer Center Sound database design can save hours of development time and ensure functionality and reliability. Database Design for Mere Mortals(TM), Second Edition, is a straightforward, platform-independent tutorial on the basic principles of relational database design. It provides a commonsense design methodology for developing databases that work. Database design expert Michael J. Hernandez has expanded his best-selling first edition, maintaining its hands-on approach and accessibility while updating its coverage and including even more examples and illustrations. This edition features a CD-ROM that includes diagrams of sample databases, as well as design guidelines, documentation forms, and examples of the database design process. This book will give you the knowledge and tools you need to create efficient and effective relational databases.

Recipes for Continuous Database Integration - Pramod J. Sadalage 2003-03-26

This is the eBook version of the printed book. The past few years have seen the rise of agile or evolutionary methods in software development. These methods embrace change in requirements even late in the project. The ability to change software is because of certain practices that are followed within teams, such as Test Driven Development, Pair Programming, and Continuous Integration. Continuous Integration provides a way for software teams to integrate their work more than once a day, and promotes confidence in the software that is being developed by the team. It is thought that this practice is difficult to apply when continuously integrating the database with application code; hence, Evolutionary Database Development is considered a mismatch with agile methods. Pramod Sadalage shows that this is not necessarily true. Continuous Integration changed the way software is written. Why not extend and make the database part of the same Continuous Integration cycle so that you can see integrated results of your application as well as your database? Delivered in PDF format for quick and easy access, Recipes for Continuous Database Integration shows how the database can be brought under the preview of Continuous Integration, allowing all teams to integrate not only their application code, but also their database. This Short Cut presents a recipe for each task that needs to be done. Each recipe starts with a statement of a problem, followed by an explanation and solution. It provides concrete ways and examples to implement ideas in Refactoring Databases: Evolutionary Database Design by Scott W Ambler and Pramod Sadalage. Table of Contents What This Short Cut Covers Introduction Recipe 1 Continuously Integrating? Recipe 2 Extracting Your Database in Scripts Recipe 3 Using Version Control for Your Database Recipe 4 Automating Database or Schema Creation Recipe 5 Creating Objects in Your Database Recipe 6 Removing Database Objects Recipe 7 Removing Your Database Recipe 8 Using the Build Property Files Recipe 9 Re-Creating Your Application Database for Any Build Recipe 10 Making It Easy for New Developers to Join the Team Recipe 11 Integrating on Every Check-In Recipe 12 Naming Upgrade Scripts Recipe 13 Automating Database Change Script Creation Recipe 14 Implementing Database Version Checking Recipe 15 Sending Upgrades to Customers Sample Code Further Reading About the Author What's in the Companion Book Related Publication

Modernizing IBM i Applications from the Database up to the User Interface and Everything in Between - Nadir K. Amra 2015-12-07

This IBM® Redbooks® publication is focused on melding industry preferred practices with the unique needs of the IBM i community and providing a holistic view of modernization. This book covers key trends for application structure, user interface, data access, and the database. Modernization is a broad term when applied to applications. It is more than a single event. It is a sequence of actions. But even more, it is a process of rethinking how to approach the creation and maintenance of applications. There are tangible

deliveries when it comes to modernization, the most notable being a modern user interface (UI), such as a web browser or being able to access applications from a mobile device. The UI, however, is only the beginning. There are many more aspects to modernization. Using modern tools and methodologies can significantly improve productivity and reduce long-term cost while positioning applications for the next decade. It is time to put the past away. Tools and methodologies have undergone significant transformation, improving functionality, usability, and productivity. This is true of the plethora of IBM tools and the wealth of tools available from many Independent Solution Providers (ISVs). This publication is the result of work that was done by IBM, industry experts, and by representatives from many of the ISV Tool Providers. Some of their tools are referenced in the book. In addition to reviewing technologies based on context, there is an explanation of why modernization is important and a description of the business benefits of investing in modernization. This critical information is key for line-of-business executives who want to understand the benefits of a modernization project. This book is appropriate for CIOs, architects, developers, and business leaders. Related information Making the Case for Modernization, IBM Systems Magazine *Beyond Software Architecture* - Luke Hohmann 2003-01

This text aims to help all members of the development team make the correct nuts-and-bolts architecture decisions that ensure project success.

Refactoring Databases - Scott W. Ambler 2006

"This comprehensive guide and reference helps you overcome the practical obstacles to refactoring real-world databases by covering every fundamental concept underlying database refactoring. Using start-to-finish examples, the authors walk you through refactoring simple standalone database applications as well as sophisticated multi-application scenarios. You'll master every task involved in refactoring database schemas, and discover best practices for deploying refactorings in even the most complex production environments."--Jacket.

Building Evolutionary Architectures - Neal Ford 2017-09-18

The software development ecosystem is constantly changing, providing a constant stream of new tools, frameworks, techniques, and paradigms. Over the past few years, incremental developments in core engineering practices for software development have created the foundations for rethinking how architecture changes over time, along with ways to protect important architectural characteristics as it evolves. This practical guide ties those parts together with a new way to think about architecture and time.

Graph Databases - Ian Robinson 2013-06-10

Discover how graph databases can help you manage and query highly connected data. With this practical book, you'll learn how to design and implement a graph database that brings the power of graphs to bear on a broad range of problem domains. Whether you want to speed up your response to user queries or build a database that can adapt as your business evolves, this book shows you how to apply the schema-free graph model to real-world problems. Learn how different organizations are using graph databases to outperform their competitors. With this book's data modeling, query, and code examples, you'll quickly be able to implement your own solution. Model data with the Cypher query language and property graph model Learn best practices and common pitfalls when modeling with graphs Plan and implement a graph database solution in test-driven fashion Explore real-world examples to learn how and why organizations use a graph database Understand common patterns and components of graph database architecture Use analytical techniques and algorithms to mine graph database information

Database Systems - Nenad Jukic 2013-01-03

An introductory, yet comprehensive, database textbook intended for use in undergraduate and graduate information systems database courses. This text also provides practical content to current and aspiring information systems, business data analysis, and decision support industry professionals. Database Systems: Introduction to Databases and Data Warehouses covers both analytical and operations database as knowledge of both is integral to being successful in today's business environment. It also provides a solid theoretical foundation and hands-on practice using an integrated web-based data-modeling suite.

Mastering PostgreSQL 12 - Hans-Jürgen Schönig 2019-11-29

Master PostgreSQL 12 features such as advanced indexing, high availability, monitoring, and much more to efficiently manage and maintain your database Key Features Grasp advanced PostgreSQL 12 concepts with

real-world examples and sample datasets Explore query parallelism, data replication, database administration, and more Extend PostgreSQL functionalities to suit your organization's needs with minimal effort Book Description Thanks to its reliability, robustness, and high performance, PostgreSQL has become the most advanced open source database on the market. This third edition of Mastering PostgreSQL helps you build dynamic database solutions for enterprise applications using the latest release of PostgreSQL, which enables database analysts to design both physical and technical aspects of system architecture with ease. Starting with an introduction to the newly released features in PostgreSQL 12, this book will help you build efficient and fault-tolerant PostgreSQL applications. You'll thoroughly examine the advanced features of PostgreSQL, including logical replication, database clusters, performance tuning, monitoring, and user management. You'll also work with the PostgreSQL optimizer, configure PostgreSQL for high speed, and understand how to move from Oracle to PostgreSQL. As you progress through the chapters, you'll cover transactions, locking, indexes, and how to optimize queries for improved performance. Additionally, you'll learn how to manage network security and explore backups and replications while understanding useful PostgreSQL extensions to help you in optimizing the performance of large databases. By the end of this PostgreSQL book, you'll be able to get the most out of your database by implementing advanced administrative tasks effortlessly. What you will learn Understand the advanced SQL functions in PostgreSQL 12 Use indexing features in PostgreSQL to fine-tune the performance of queries Work with stored procedures and manage backup and recovery Master replication and failover techniques to reduce data loss Replicate PostgreSQL database systems to create backups and to scale your database Manage and improve the security of your server to protect your data Troubleshoot your PostgreSQL instance for solutions to common and not-so-common problems Who this book is for This book is for PostgreSQL developers and administrators and database professionals who want to implement advanced functionalities and master complex administrative tasks with PostgreSQL 12. Prior exposure to PostgreSQL as well as familiarity with the basics of database administration is expected.

User Stories Applied - Mike Cohn 2004-03-01

Thoroughly reviewed and eagerly anticipated by the agile community, User Stories Applied offers a requirements process that saves time, eliminates rework, and leads directly to better software. The best way to build software that meets users' needs is to begin with "user stories": simple, clear, brief descriptions of functionality that will be valuable to real users. In User Stories Applied, Mike Cohn provides you with a front-to-back blueprint for writing these user stories and weaving them into your development lifecycle. You'll learn what makes a great user story, and what makes a bad one. You'll discover practical ways to gather user stories, even when you can't speak with your users. Then, once you've compiled your user stories, Cohn shows how to organize them, prioritize them, and use them for planning, management, and testing. User role modeling: understanding what users have in common, and where they differ Gathering stories: user interviewing, questionnaires, observation, and workshops Working with managers, trainers, salespeople and other "proxies" Writing user stories for acceptance testing Using stories to prioritize, set schedules, and estimate release costs Includes end-of-chapter practice questions and exercises User Stories Applied will be invaluable to every software developer, tester, analyst, and manager working with any agile method: XP, Scrum... or even your own home-grown approach.

Advanced Information Systems Engineering - Schahram Dustdar 2020-06-02

This book constitutes the refereed proceedings of the 32nd International Conference on Advanced Information Systems Engineering, CAiSE 2020, held in Grenoble, France, in June 2020.* The 33 full papers presented in this volume were carefully reviewed and selected from 185 submissions. The book also contains one invited talk in full paper length. The papers were organized in topical sections named: distributed applications; AI and big data in IS; process mining and analysis; requirements and modeling; and information systems engineering. Abstracts on the CAiSE 2020 tutorials can be found in the back matter of the volume. *The conference was held virtually due to the COVID-19 pandemic.

NoSQL Distilled - Pramod J. Sadalage 2013

'NoSQL Distilled' is designed to provide you with enough background on how NoSQL databases work, so that you can choose the right data store without having to trawl the whole web to do it. It won't answer your questions definitively, but it should narrow down the range of options you have to consider.

Refactoring Workbook - William C. Wake 2004

& Most software practitioners deal with inherited code; this book teaches them how to optimize it & & Workbook approach facilitates the learning process & & Helps you identify where problems in a software application exist or are likely to exist

Agile Modeling - Scott Ambler 2002-08-14

The first book to cover Agile Modeling, a new modeling technique created specifically for XP projects eXtreme Programming (XP) has created a buzz in the software development community-much like Design Patterns did several years ago. Although XP presents a methodology for faster software development, many developers find that XP does not allow for modeling time, which is critical to ensure that a project meets its proposed requirements. They have also found that standard modeling techniques that use the Unified Modeling Language (UML) often do not work with this methodology. In this innovative book, Software Development columnist Scott Ambler presents Agile Modeling (AM)-a technique that he created for modeling XP projects using pieces of the UML and Rational's Unified Process (RUP). Ambler clearly explains AM, and shows readers how to incorporate AM, UML, and RUP into their development projects with the help of numerous case studies integrated throughout the book. AM was created by the author for modeling XP projects-an element lacking in the original XP design The XP community and its creator have embraced AM, which should give this book strong market acceptance Companion Web site at www.agilemodeling.com features updates, links to XP and AM resources, and ongoing case studies about agile modeling.

Database and Expert Systems Applications - Sven Hartmann 2016-08-05

This two volume set LNCS 9827 and LNCS 9828 constitutes the refereed proceedings of the 27th International Conference on Database and Expert Systems Applications, DEXA 2016, held in Porto, Portugal, September 2016. The 39 revised full papers presented together with 29 short papers were carefully reviewed and selected from 137 submissions. The papers discuss a range of topics including: Temporal, Spatial, and High Dimensional Databases; Data Mining; Authenticity, Privacy, Security, and Trust; Data Clustering; Distributed and Big Data Processing; Decision Support Systems, and Learning; Data Streams; Data Integration, and Interoperability; Semantic Web, and Data Semantics; Social Networks, and Network Analysis; Linked Data; Data Analysis; NoSQL, NewSQL; Multimedia Data; Personal Information Management; Semantic Web and Ontologies; Database and Information System Architectures; Query Answering and Optimization; Information Retrieval, and Keyword Search; Data Modelling, and Uncertainty.

Continuous Integration - Paul M. Duvall 2007-06-29

For any software developer who has spent days in "integration hell," cobbling together myriad software components, Continuous Integration: Improving Software Quality and Reducing Risk illustrates how to transform integration from a necessary evil into an everyday part of the development process. The key, as the authors show, is to integrate regularly and often using continuous integration (CI) practices and techniques. The authors first examine the concept of CI and its practices from the ground up and then move on to explore other effective processes performed by CI systems, such as database integration, testing, inspection, deployment, and feedback. Through more than forty CI-related practices using application examples in different languages, readers learn that CI leads to more rapid software development, produces deployable software at every step in the development lifecycle, and reduces the time between defect introduction and detection, saving time and lowering costs. With successful implementation of CI, developers reduce risks and repetitive manual processes, and teams receive better project visibility. The book covers How to make integration a "non-event" on your software development projects How to reduce the amount of repetitive processes you perform when building your software Practices and techniques for using CI effectively with your teams Reducing the risks of late defect discovery, low-quality software, lack of visibility, and lack of deployable software Assessments of different CI servers and related tools on the market The book's companion Web site, www.integratebutton.com, provides updates and code examples.

Refactoring Databases - Scott W. Ambler 2006-03-03

Refactoring has proven its value in a wide range of development projects-helping software professionals improve system designs, maintainability, extensibility, and performance. Now, for the first time, leading agile methodologist Scott Ambler and renowned consultant Pramodkumar Sadalage introduce powerful

refactoring techniques specifically designed for database systems. Ambler and Sadalage demonstrate how small changes to table structures, data, stored procedures, and triggers can significantly enhance virtually any database design-without changing semantics. You'll learn how to evolve database schemas in step with source code-and become far more effective in projects relying on iterative, agile methodologies. This comprehensive guide and reference helps you overcome the practical obstacles to refactoring real-world databases by covering every fundamental concept underlying database refactoring. Using start-to-finish examples, the authors walk you through refactoring simple standalone database applications as well as sophisticated multi-application scenarios. You'll master every task involved in refactoring database schemas, and discover best practices for deploying refactorings in even the most complex production environments. The second half of this book systematically covers five major categories of database refactorings. You'll learn how to use refactoring to enhance database structure, data quality, and referential integrity; and how to refactor both architectures and methods. This book provides an extensive set of examples built with Oracle and Java and easily adaptable for other languages, such as C#, C++, or VB.NET, and other databases, such as DB2, SQL Server, MySQL, and Sybase. Using this book's techniques and examples, you can reduce waste, rework, risk, and cost-and build database systems capable of evolving smoothly, far into the future.

Building Microservices - Sam Newman 2015-02-02

Annotation Over the past 10 years, distributed systems have become more fine-grained. From the large multi-million line long monolithic applications, we are now seeing the benefits of smaller self-contained services. Rather than heavy-weight, hard to change Service Oriented Architectures, we are now seeing systems consisting of collaborating microservices. Easier to change, deploy, and if required retire, organizations which are in the right position to take advantage of them are yielding significant benefits. This book takes an holistic view of the things you need to be cognizant of in order to pull this off. It covers just enough understanding of technology, architecture, operations and organization to show you how to move towards finer-grained systems.

Refactoring - Jay Fields 2009-10

The Definitive Refactoring Guide, Fully Revamped for Ruby With refactoring, programmers can transform even the most chaotic software into well-designed systems that are far easier to evolve and maintain. What's more, they can do it one step at a time, through a series of simple, proven steps. Now, there's an authoritative and extensively updated version of Martin Fowler's classic refactoring book that utilizes Ruby examples and idioms throughout-not code adapted from Java or any other environment. The authors introduce a detailed catalog of more than 70 proven Ruby refactorings, with specific guidance on when to apply each of them, step-by-step instructions for using them, and example code illustrating how they work. Many of the authors' refactorings use powerful Ruby-specific features, and all code samples are available for download. Leveraging Fowler's original concepts, the authors show how to perform refactoring in a controlled, efficient, incremental manner, so you methodically improve your code's structure without introducing new bugs. Whatever your role in writing or maintaining Ruby code, this book will be an indispensable resource. This book will help you * Understand the core principles of refactoring and the reasons for doing it * Recognize "bad smells" in your Ruby code * Rework bad designs into well-designed code, one step at a time * Build tests to make sure your refactorings work properly * Understand the challenges of refactoring and how they can be overcome * Compose methods to package code properly * Move features between objects to place responsibilities where they fit best * Organize data to make it easier to work with * Simplify conditional expressions and make more effective use of polymorphism * Create interfaces that are easier to understand and use * Generalize more effectively * Perform larger refactorings that transform entire software systems and may take months or years * Successfully refactor Ruby on Rails code

The Object Primer - Scott W. Ambler 2004-03-22

The acclaimed beginner's book on object technology now presents UML 2.0, Agile Modeling, and object development techniques.

Disciplined Agile Delivery - Scott W. Ambler 2012-05-31

Master IBM's Breakthrough DAD Process Framework for Succeeding with Agile in Large, Complex,

Mission-Critical IT Projects It is widely recognized that moving from traditional to agile approaches to build software solutions is a critical source of competitive advantage. Mainstream agile approaches that are indeed suitable for small projects require significant tailoring for larger, complex enterprise projects. In *Disciplined Agile Delivery*, Scott W. Ambler and Mark Lines introduce IBM's breakthrough Disciplined Agile Delivery (DAD) process framework, which describes how to do this tailoring. DAD applies a more disciplined approach to agile development by acknowledging and dealing with the realities and complexities of a portfolio of interdependent program initiatives. Ambler and Lines show how to extend Scrum with supplementary agile and lean strategies from Agile Modeling (AM), Extreme Programming (XP), Kanban, Unified Process (UP), and other proven methods to provide a hybrid approach that is adaptable to your organization's unique needs. They candidly describe what practices work best, why they work, what the trade-offs are, and when to consider alternatives, all within the context of your situation. *Disciplined Agile Delivery* addresses agile practices across the entire lifecycle, from requirements, architecture, and development to delivery and governance. The authors show how these best-practice techniques fit together in an end-to-end process for successfully delivering large, complex systems--from project initiation through delivery. Coverage includes Scaling agile for mission-critical enterprise endeavors Avoiding mistakes that drive poorly run agile projects to chaos Effectively initiating an agile project Transitioning as an individual to agile Incrementally building consumable solutions Deploying agile solutions into complex production environments Leveraging DevOps, architecture, and other enterprise disciplines Adapting your governance strategy for agile projects Based on facts, research, and extensive

experience, this book will be an indispensable resource for every enterprise software leader and practitioner--whether they're seeking to optimize their existing agile/Scrum process or improve the agility of an iterative process.

Monolith to Microservices - Sam Newman 2019-11-14

How do you detangle a monolithic system and migrate it to a microservice architecture? How do you do it while maintaining business-as-usual? As a companion to Sam Newman's extremely popular *Building Microservices*, this new book details a proven method for transitioning an existing monolithic system to a microservice architecture. With many illustrative examples, insightful migration patterns, and a bevy of practical advice to transition your monolith enterprise into a microservice operation, this practical guide covers multiple scenarios and strategies for a successful migration, from initial planning all the way through application and database decomposition. You'll learn several tried and tested patterns and techniques that you can use as you migrate your existing architecture. Ideal for organizations looking to transition to microservices, rather than rebuild Helps companies determine whether to migrate, when to migrate, and where to begin Addresses communication, integration, and the migration of legacy systems Discusses multiple migration patterns and where they apply Provides database migration examples, along with synchronization strategies Explores application decomposition, including several architectural refactoring patterns Delves into details of database decomposition, including the impact of breaking referential and transactional integrity, new failure modes, and more

The Elements of UML(TM) 2.0 Style - Scott W. Ambler 2005-05-09

Concise and easy-to-understand guidelines and standards for creating UML 2.0 diagrams.