Site To Download Rapid Development Developer Best Practices

Thank you categorically much for downloading **Rapid Development Developer Best Practices**.Maybe you have knowledge that, people have look numerous time for their favorite books in the same way as this Rapid Development Developer Best Practices, but end occurring in harmful downloads.

Rather than enjoying a fine book once a cup of coffee in the afternoon, otherwise they juggled later than some harmful virus inside their computer. **Rapid Development Developer Best Practices** is approachable in our digital library an online admission to it is set as public thus you can download it instantly. Our digital library saves in combination countries, allowing you to acquire the most less latency era to download any of our books subsequently this one. Merely said, the Rapid Development Developer Best Practices is universally compatible in imitation of any devices to read.

KEY=BEST - ELVIS AVILA

Rapid Development Taming Wild Software Schedules *Pearson Education* **Project managers, technical leads, and Windows programmers throughout the industry share an important concern--how to get their development schedules under control. Rapid Development addresses that concern head-on with philosophy, techniques, and tools that help shrink and control development schedules and keep projects moving. The style is friendly and conversational--and the content is impressive. Software Project Survival Guide** *Pearson Education* **Looks at a successful software project and provides details for software development for clients using object-oriented design and programming. Code Complete** *Pearson Education* **Widely considered one of the best practical guides to programming, Steve McConnell's original CODE COMPLETE has been helping developers write better software for more than a decade. Now this classic book has been fully updated and revised with leading-edge practices—and hundreds of new code samples—illustrating the art and science of software construction. Capturing the body of knowledge available from research, academia, and everyday commercial practice, McConnell synthesizes the most effective techniques and must-know principles into clear, pragmatic guidance. No matter what your experience level, development environment, or project size, this book will inform and stimulate your thinking—and help you build the highest quality code. Discover the timeless techniques and strategies that help you: Design for minimum complexity and maximum creativity Reap the benefits of collaborative development Apply defensive programming techniques to reduce and flush out errors Exploit opportunities to refactor—or evolve—code, and do it safely Use construction practices that are right-weight for your project Debug problems quickly and effectively Resolve critical construction issues early and correctly Build quality into the beginning, middle, and end of your project Writing Secure Code** *Pearson Education* **Covers topics such as the importance of secure systems, threat modeling, canonical representation issues, solving database input, denial-of-service attacks, and security code reviews and checklists. Software Estimation Demystifying the Black Art** *Microsoft Press* **Often referred to as the "black art" because of its complexity and uncertainty, software estimation is not as difficult or puzzling as people think. In fact, generating accurate estimates is straightforward—once you understand the art of creating them. In his highly anticipated book, acclaimed author Steve McConnell unravels the mystery to successful software estimation—distilling academic information and real-world experience into a practical guide for working software professionals. Instead of arcane treatises and rigid modeling techniques, this guide highlights a proven set of procedures, understandable formulas, and heuristics that individuals and development teams can apply to their projects to help achieve estimation proficiency. Discover how to: Estimate schedule and cost—or estimate the functionality that can be delivered within a given time frame Avoid common software estimation mistakes Learn estimation techniques for you, your team, and your organization * Estimate specific project activities—including development, management, and defect correction Apply estimation approaches to any type of project—small or large, agile or traditional Navigate the shark-infested political waters that surround project estimates When many corporate software projects are failing, McConnell shows you what works for successful software estimation. Solid Code** *Microsoft Press* **Get best-in-class engineering practices to help you write more-robust, bug-free code. Two Microsoft .NET development experts share real-world examples and proven methods for optimizing the software development life cycle—from avoiding costly programming pitfalls to making your development team more efficient. Managed code developers at all levels will find design, prototyping, implementation, debugging, and testing tips to boost the quality of their code—today. Optimize each stage of the development process—from design to testing—and produce higher-quality applications. Use metaprogramming to reduce code complexity, while increasing flexibility and maintainability Treat performance as a feature—and manage it throughout the development life cycle Apply best practices for application scalability Employ preventative security measures to ward off malicious attacks Practice defensive programming to catch bugs before run time Incorporate automated builds, code analysis, and testing into the daily engineering process Implement better source-control management and check-in procedures Establish a quality-driven, milestone-based project rhythm—and improve your results! Adaptive Code Agile coding with design patterns and SOLID principles** *Microsoft Press* **Write code that can adapt to changes. By applying this book's principles, you can create code that accommodates new requirements and unforeseen scenarios without significant rewrites. Gary McLean Hall describes Agile best practices, principles, and patterns for designing and writing code that can evolve more quickly and easily, with fewer**

errors, because it doesn't impede change. Now revised, updated, and expanded, Adaptive Code, Second Edition adds indispensable practical insights on Kanban, dependency inversion, and creating reusable abstractions. Drawing over a decade of Agile consulting and development experience, McLean Hall has updated his best-seller with deeper coverage of unit testing, refactoring, pure dependency injection, and more. Master powerful new ways to: • Write code that enables and complements Scrum, Kanban, or any other Agile framework • Develop code that can survive major changes in requirements • Plan for adaptability by using dependencies, layering, interfaces, and design patterns • Perform unit testing and refactoring in tandem, gaining more value from both • Use the "golden master" technique to make legacy code adaptive • Build SOLID code with single-responsibility, open/closed, and Liskov substitution principles • Create smaller interfaces to support more-diverse client and architectural needs • Leverage dependency injection best practices to improve code adaptability • Apply dependency inversion with the Stairway pattern, and avoid related anti-patterns About You This book is for programmers of all skill levels seeking more-practical insight into design patterns, SOLID principles, unit testing, refactoring, and related topics. Most readers will have programmed in C#, Java, C++, or similar object-oriented languages, and will be familiar with core procedural programming techniques. Practices of an Agile Developer Working in the Real World *Pragmatic Bookshelf* These are the proven, effective agile practices that will make you a better developer. You'll learn pragmatic ways of approaching the development process and your personal coding techniques. You'll learn about your own attitudes, issues with working on a team, and how to best manage your learning, all in an iterative, incremental, agile style. You'll see how to apply each practice, and what benefits you can expect. Bottom line: This book will make you a better developer. Rapid Development *Microsoft Press* Corporate and commercial software-development teams all want solutions for one important problem—how to get their high-pressure development schedules under control. In RAPID DEVELOPMENT, author Steve McConnell addresses that concern head-on with overall strategies, specific best practices, and valuable tips that help shrink and control development schedules and keep projects moving. Inside, you'll find: A rapid-development strategy that can be applied to any project and the best practices to make that strategy work Candid discussions of great and not-so-great rapid-development practices—estimation, prototyping, forced overtime, motivation, teamwork, rapid-development languages, risk management, and many others A list of classic mistakes to avoid for rapid-development projects, including creeping requirements, shortchanged quality, and silver-bullet syndrome Case studies that vividly illustrate what can go wrong, what can go right, and how to tell which direction your project is going RAPID DEVELOPMENT is the real-world guide to more efficient applications development. Pro .NET Best Practices *Apress* Pro .NET Best Practices is a practical reference to the best practices that you can apply to your .NET projects today. You will learn standards, techniques, and conventions that are sharply focused, realistic and helpful for achieving results, steering clear of unproven, idealistic, and impractical recommendations. Pro .NET Best Practices covers a broad range of practices and principles that development experts agree are the right ways to develop software, which includes continuous integration, automated testing, automated deployment, and code analysis. Whether the solution is from a free and open source or a commercial offering, you will learn how to get a continuous integration server running and executing builds every time code changes. You will write clearer and more maintainable automated testing code that focuses on prevention and helping your .NET project succeed. By learning and following the .NET best practices in this book, you will avoid making the same mistakes once. With this book at your side, you'll get: Real-world, no-nonsense approaches to continuous integration, automated testing, automated deployment, and code analysis Tips and tricks you'll need to clear hurdles that keep others from putting these common sense ideas into common practice Guidance from the minimal, essential approach all the way to what's necessary to deliver at the highest levels of quality and effectiveness Benefit immediately, even before finishing it, from the knowledge, workable advice, and experience found in Pro .NET Best Practices. MOS Study Guide for Microsoft Access Expert Exam MO-500 *Microsoft Press* Advance your everyday proficiency with Access 2019. And earn the credential that proves it! Demonstrate your expertise with Microsoft Access! Designed to help you practice and prepare for Microsoft Office Specialist (MOS): Access 2019 certification, this official Study Guide delivers: In-depth preparation for each MOS objective Detailed procedures to help build the skills measured by the exam Hands-on tasks to practice what you've learned Practice files and sample solutions Sharpen the skills measured by these objectives: Create and manage databases Build tables Create queries Create forms Create reports About MOS A Microsoft Office Specialist (MOS) certification validates your proficiency with Microsoft Office programs, demonstrating that you can meet globally recognized performance standards. Hands-on experience with the technology is required to successfully pass Microsoft Certification exams. Building Software Teams Ten Best Practices for Effective Software Development *"O'Reilly Media, Inc."* Why does poor software quality continue to plague enterprises of all sizes in all industries? Part of the problem lies with the process, rather than individual developers. This practical guide provides ten best practices to help team leaders create an effective working environment through key adjustments to their process. As a follow-up to their popular book, Building Maintainable Software, consultants with the Software Improvement Group (SIG) offer critical lessons based on their assessment of development processes used by hundreds of software teams. Each practice includes examples of goalsetting to help you choose the right metrics for your team. Achieve development goals by determining meaningful metrics with the Goal-Question-Metric approach Translate those goals to a verifiable Definition of Done Manage code versions for consistent and predictable modification Control separate environments for each stage in the development pipeline Automate tests as much as possible and steer their guidelines and expectations Let the Continuous Integration server do much of the hard work for you Automate the process of pushing code through the pipeline Define development process standards to improve consistency and simplicity Manage dependencies on third party code to keep your software consistent and up to date Document only the most necessary and

current knowledge Rapid Application Development with OutSystems Create applications with OutSystems up to seven times faster than with traditional technologies *Packt Publishing Ltd* Discover best practices to create simple, robust, and scalable software solutions quickly and securely, and focus on what's important - the business and the end customer Key Features Discover strategies and tips for low-code and no-code development with the help of useful examples Build enterprise-scale reactive web and mobile applications Explore the benefits of low-code development Book Description OutSystems is a software development platform that speeds up the build phase by abstracting code and making almost everything visual. This means replacing textual language with visual artifacts that avoid lexical errors and speed up code composition using accelerators and predefined templates. The book begins by walking you through the fundamentals of the technology, along with a general overview of end-to-end web and mobile software development. You'll learn how to configure your personal area in the cloud and use the OutSystems IDE to connect with it. The book then shows you how to build a web application based on the best architectural and developmental practices in the market, and takes the same approach for the mobile paradigm. As you advance, you'll find out how to develop the same application, and the great potential of reusing code from one paradigm in another and the symbiosis between them is showcased.The only application that'll differ from the application in the exercise is the one used in business process technology (BPT), with a focus on a common market use case. By the end of this OutSystems book, you'll be able to develop enterprise-level applications on the web and mobile, integrating them with third parties and other systems on the market. You'll also understand the concepts of performance, security, and software construction and be able to apply them effectively. What you will learn Get to grips with the OutSystems infrastructure Develop reactive web and mobile applications in OutSystems using best practices in Service Studio Define and design Architecture Canvas (3-Layer Canvas) for your applications Integrate with external systems in OutSystems Enhance applications with BPT Manage and monitor your applications in the OutSystems infrastructure Who this book is for This book is for back-end developers, tech leaders, UX/UI developers, frontend developers, (in general, full-stack developers), tech companies, and enterprises looking to learn how to develop web and mobile software quickly and disruptively by leveraging OutSystems, one of the most low-code platforms in the market. An IT background is not mandatory; however, experience in SQL, JavaScript, HTML, CSS, and C# is required to get started with this book. The Nature of Software Development Keep It Simple, Make It Valuable, Build It Piece by Piece *Pragmatic Bookshelf* You need to get value from your software project. You need it "free, now, and perfect." We can't get you there, but we can help you get to "cheaper, sooner, and better." This book leads you from the desire for value down to the specific activities that help good Agile projects deliver better software sooner, and at a lower cost. Using simple sketches and a few words, the author invites you to follow his path of learning and understanding from a half century of software development and from his engagement with Agile methods from their very beginning. The book describes software development, starting from our natural desire to get something of value. Each topic is described with a picture and a few paragraphs. You're invited to think about each topic; to take it in. You'll think about how each step into the process leads to the next. You'll begin to see why Agile methods ask for what they do, and you'll learn why a shallow implementation of Agile can lead to only limited improvement. This is not a detailed map, nor a step-by-step set of instructions for building the perfect project. There is no map or instructions that will do that for you. You need to build your own project, making it a bit more perfect every day. To do that effectively, you need to build up an understanding of the whole process. This book points out the milestones on your journey of understanding the nature of software development done well. It takes you to a location, describes it briefly, and leaves you to explore and fill in your own understanding. What You Need: You'll need your Standard Issue Brain, a bit of curiosity, and a desire to build your own understanding rather than have someone else's detailed ideas poured into your head. Beautiful Code Leading Programmers Explain How They Think "*O'Reilly Media, Inc.*" How do the experts solve difficult problems in software development? In this unique and insightful book, leading computer scientists offer case studies that reveal how they found unusual, carefully designed solutions to high-profile projects. You will be able to look over the shoulder of major coding and design experts to see problems through their eyes. This is not simply another design patterns book, or another software engineering treatise on the right and wrong way to do things. The authors think aloud as they work through their project's architecture, the tradeoffs made in its construction, and when it was important to break rules. This book contains 33 chapters contributed by Brian Kernighan, KarlFogel, Jon Bentley, Tim Bray, Elliotte Rusty Harold, Michael Feathers,Alberto Savoia, Charles Petzold, Douglas Crockford, Henry S. Warren,Jr., Ashish Gulhati, Lincoln Stein, Jim Kent, Jack Dongarra and PiotrLuszczek, Adam Kolawa, Greg Kroah-Hartman, Diomidis Spinellis, AndrewKuchling, Travis E. Oliphant, Ronald Mak, Rogerio Atem de Carvalho andRafael Monnerat, Bryan Cantrill, Jeff Dean and Sanjay Ghemawat, SimonPeyton Jones, Kent Dybvig, William Otte and Douglas C. Schmidt, AndrewPatzer, Andreas Zeller, Yukihiro Matsumoto, Arun Mehta, TV Raman,Laura Wingerd and Christopher Seiwald, and Brian Hayes. Beautiful Code is an opportunity for master coders to tell their story. All author royalties will be donated to Amnesty International. Pro .NET Best Practices *Apress* Pro .NET Best Practices is a practical reference to the best practices that you can apply to your .NET projects today. You will learn standards, techniques, and conventions that are sharply focused, realistic and helpful for achieving results, steering clear of unproven, idealistic, and impractical recommendations. Pro .NET Best Practices covers a broad range of practices and principles that development experts agree are the right ways to develop software, which includes continuous integration, automated testing, automated deployment, and code analysis. Whether the solution is from a free and open source or a commercial offering, you will learn how to get a continuous integration server running and executing builds every time code changes. You will write clearer and more maintainable automated testing code that focuses on prevention and helping your .NET project succeed. By learning and following the .NET best practices in this book, you will avoid making the same mistakes

once. With this book at your side, you'll get: Real-world, no-nonsense approaches to continuous integration, automated testing, automated deployment, and code analysis Tips and tricks you'll need to clear hurdles that keep others from putting these common sense ideas into common practice Guidance from the minimal, essential approach all the way to what's necessary to deliver at the highest levels of quality and effectiveness Benefit immediately, even before finishing it, from the knowledge, workable advice, and experience found in Pro .NET Best Practices. Code Complete, 2nd Edition Widely considered one of the best practical guides to programming, Steve McConnell s original CODE COMPLETE has been helping developers write better software for more than a decade. Now this classic book has been fully updated and revised with leading-edge practices-and hundreds of new code samples-illustrating the art and science of software construction. Capturing the body of knowledge available from research, academia, and everyday commercial practice, McConnell synthesizes the most effective techniques and must-know principles into clear, pragmatic guidance. No matter what your experience level, development environment, or project size, this book will inform and stimulate your thinking-and help you build the highest quality code. The Pragmatic Programmer From Journeyman to Master *Addison-Wesley Professional* What others in the trenches say about The Pragmatic Programmer... "The cool thing about this book is that it's great for keeping the programming process fresh. The book helps you to continue to grow and clearly comes from people who have been there." —Kent Beck, author of Extreme Programming Explained: Embrace Change "I found this book to be a great mix of solid advice and wonderful analogies!" —Martin Fowler, author of Refactoring and UML Distilled "I would buy a copy, read it twice, then tell all my colleagues to run out and grab a copy. This is a book I would never loan because I would worry about it being lost." —Kevin Ruland, Management Science, MSG-Logistics "The wisdom and practical experience of the authors is obvious. The topics presented are relevant and useful.... By far its greatest strength for me has been the outstanding analogies—tracer bullets, broken windows, and the fabulous helicopter-based explanation of the need for orthogonality, especially in a crisis situation. I have little doubt that this book will eventually become an excellent source of useful information for journeymen programmers and expert mentors alike." —John Lakos, author of Large-Scale C++ Software Design "This is the sort of book I will buy a dozen copies of when it comes out so I can give it to my clients." —Eric Vought, Software Engineer "Most modern books on software development fail to cover the basics of what makes a great software developer, instead spending their time on syntax or technology where in reality the greatest leverage possible for any software team is in having talented developers who really know their craft well. An excellent book." —Pete McBreen, Independent Consultant "Since reading this book, I have implemented many of the practical suggestions and tips it contains. Across the board, they have saved my company time and money while helping me get my job done quicker! This should be a desktop reference for everyone who works with code for a living." —Jared Richardson, Senior Software Developer, iRenaissance, Inc. "I would like to see this issued to every new employee at my company...." —Chris Cleeland, Senior Software Engineer, Object Computing, Inc. "If I'm putting together a project, it's the authors of this book that I want. . . . And failing that I'd settle for people who've read their book." —Ward Cunningham Straight from the programming trenches, The Pragmatic Programmer cuts through the increasing specialization and technicalities of modern software development to examine the core process--taking a requirement and producing working, maintainable code that delights its users. It covers topics ranging from personal responsibility and career development to architectural techniques for keeping your code flexible and easy to adapt and reuse. Read this book, and you'll learn how to Fight software rot; Avoid the trap of duplicating knowledge; Write flexible, dynamic, and adaptable code; Avoid programming by coincidence; Bullet-proof your code with contracts, assertions, and exceptions; Capture real requirements; Test ruthlessly and effectively; Delight your users; Build teams of pragmatic programmers; and Make your developments more precise with automation. Written as a series of self-contained sections and filled with entertaining anecdotes, thoughtful examples, and interesting analogies, The Pragmatic Programmer illustrates the best practices and major pitfalls of many different aspects of software development. Whether you're a new coder, an experienced programmer, or a manager responsible for software projects, use these lessons daily, and you'll quickly see improvements in personal productivity, accuracy, and job satisfaction. You'll learn skills and develop habits and attitudes that form the foundation for long-term success in your career. You'll become a Pragmatic Programmer. PC Mag PCMag.com is a leading authority on technology, delivering Labs-based, independent reviews of the latest products and services. Our expert industry analysis and practical solutions help you make better buying decisions and get more from technology. Professional Software Development Shorter Schedules, Higher Quality Products, More Successful Projects, Enhanced Careers *Addison-Wesley Professional* bull; Renowned software expert Steve McConnell presents his latest thoughts on the condition of the software engineering profession bull; Helps software developers regain the sight of the big-picture reasons why their jobs matter bull; A thinking man's guide to the current state of software Oracle Embedded Programming and Application Development *CRC Press* Focusing on tried and true best practice techniques in cross-technology based Oracle embedded programming, this book provides authoritative guidance for improving your code compilation and execution. Geared towards IT professionals developing Oracle-based Web-enabled applications in PL/SQL, Java, C, C++, .NET, Perl, and PHP, it covers application development from concepts to customization, following a pragmatic approach to design, coding, testing, deployment, and customization—explaining how to maximize embedded programming practices. Oracle Embedded Programming and Application Development explains application development frameworks using 3GL and 4GL high-level language code as embedded code segments across .NET, Java, and Open Source technologies, in conjunction with SQL and/or PL/SQL and the Oracle RDBMS through version 11gR2. It also: Features pluggable code using parameterized constructs to promote code reuse Explains when to use a particular embedded language as a best fit for specific applications Highlights design considerations that reduce the probability of errors, enable quick

resolution, and boost performance in terms of enabling a Fast-Actionable-Synchronized-Tested (FAST) solution implementation Provides best practice techniques that can enhance any application development code-design methodology for a better, easier, faster, cheaper, and pervasive solution that in turn helps achieve a Better Business Benefit (B-B-B) This practical guide details techniques for constructing architecture and code design methodologies for live application development projects that can be generalized and standardized as application development and code design frameworks. Cover to cover, the text provides an understanding of how the designed, developed, and deployed solutions conform to emerging and next-generation trends. It also discusses the conformance and usage of Web 2.0-based RIA functionality and regulatory compliance practices involving auditing and security. Praise for: "Taking an Oracle-centric approach, Lakshman skillfully guides you through the maze of various popular programming languages and environments including .NET, C/C++, Perl, PHP, Java, and even SQL and PL/SQL – not only showing you how they interact with Oracle but also which language is the best fit for a given situation." —John Kanagaraj, Executive Editor, IOUG SELECT Journal Introduction to Agile Methods *Pearson Education* A Thorough Introduction to the Agile Framework and Methodologies That Are Used Worldwide Organizations of all shapes and sizes are embracing Agile methodologies as a way to transform their products, customer satisfaction, and employee engagement. Many people with varying levels of work experience are interested in understanding the architecture and nuances of Agile, but it is difficult to know where to start. Numerous practitioner books are available, but there has never been a single source for unbiased information about Agile methodologies-until now. Introduction to Agile Methods is the place to start for students and professionals who want to understand Agile and become conversant with Agile values, principles, framework, and processes. Authors Sondra Ashmore and Kristin Runyan use academic research and their own experiences with numerous Agile implementations to present a clear description of the essential concepts. They address all key roles and the entire development life cycle, including common roadblocks that must be overcome to be successful. Through the authors' realistic use cases, practical examples, and thought-provoking interviews with pioneering practitioners, complex concepts are made relatable. No matter what your role or level of experience, this book provides a foundational understanding that can be used to start or enhance any Agile effort. Coverage includes How Agile compares with the Waterfall method and when to use each Why Agile demands a cultural transformation-and how that looks to each participant Comparing various Agile methodologies, including Scrum, Kanban, Extreme Programming (XP), Crystal, Feature Driven Development (FDD), Lean, and DSDM Understanding the roles within Agile and how they work together to create superior results Agile approaches to requirements gathering, planning, estimating, tracking, reporting, testing, quality, and integration Extending Agile beyond IT Site Reliability Engineering How Google Runs Production Systems *"O'Reilly Media, Inc."* The overwhelming majority of a software system's lifespan is spent in use, not in design or implementation. So, why does conventional wisdom insist that software engineers focus primarily on the design and development of large-scale computing systems? In this collection of essays and articles, key members of Google's Site Reliability Team explain how and why their commitment to the entire lifecycle has enabled the company to successfully build, deploy, monitor, and maintain some of the largest software systems in the world. You'll learn the principles and practices that enable Google engineers to make systems more scalable, reliable, and efficient—lessons directly applicable to your organization. This book is divided into four sections: Introduction—Learn what site reliability engineering is and why it differs from conventional IT industry practices Principles—Examine the patterns, behaviors, and areas of concern that influence the work of a site reliability engineer (SRE) Practices—Understand the theory and practice of an SRE's day-to-day work: building and operating large distributed computing systems Management—Explore Google's best practices for training, communication, and meetings that your organization can use Mastering BEA WebLogic Server Best Practices for Building and Deploying J2EE Applications *John Wiley & Sons* Core Internet Application Development Using ASP.NET 2.0 *Prentice Hall Professional* The Comprehensive Guide to ASP.NET 2.0 for Experienced Developers ASP.NET 2.0 represents a true breakthrough in Web development technology and delivers unprecedented power, flexibility, and efficiency. If you're an experienced programmer who wants to build production-quality Web applications and services with ASP.NET 2.0, this book is the deepest, most practical tutorial you can find. Randy Connolly introduces today's best practices for every facet of ASP.NET 2.0 development. He illuminates ASP.NET 2.0 Web server control architecture, sophisticated user interface capabilities, and navigation controls. He presents systematic, practical coverage of ASP.NET 2.0 data integration, state management, personalization, and more. As with all books in the Prentice Hall Core Series, Core Internet Application Development with ASP.NET 2.0 focuses on solving serious problems with professional-quality code. With practical insights into everything from data binding to security, this is the ASP.NET 2.0 book you've been searching for: a definitive guide to building industrial-strength Web solutions. This Book Delivers Best practices for architecting and implementing state-of-the-art Web applications and services Expert insight into the entire ASP.NET development process, from design through deployment Hundreds of professional-quality code examples, including complete applications and business objects In-depth coverage of the latest ASP.NET features, including master pages, new data source controls, and ASP.NET AJAX (a.k.a. "Atlas") Every Core Series Book Demonstrates practical techniques used by professional developers Features robust, thoroughly tested sample code and realistic examples Focuses on the cutting-edge technologies you need to master today Provides expert advice that will help you build superior software Preface Acknowledgments About the Author Part I: Core ASP.NET Chapter 1: Introducing ASP.NET 2.0 Chapter 2: How ASP.NET Works Chapter 3: Working with the Standard Web Server Controls Chapter 4: The Additional Standard Web Server Controls Chapter 5: Exception Handling and Validation Controls Chapter 6: Customizing and Managing Your Site's Appearance Chapter 7: ASP.NET Site Navigation Part II: Working with Data Chapter 8: Data Binding and Representation Chapter 9: Using ADO.NET Chapter 10: Data Controls Chapter 11: Designing and Implementing Web Applications Chapter 12: Managing ASP.NET

State Part III: Implementing Web Applications Chapter 13: Security, Membership, and Role Management 833 Chapter 15: Web Services Chapter 16: Internationalization and Deployment Appendix: ASP.NET AJAX Sneak Peek Index Pro Python Best Practices Debugging, Testing and Maintenance *Apress* Learn software engineering and coding best practices to write Python code right and error free. In this book you'll see how to properly debug, organize, test, and maintain your code, all of which leads to better, more efficient coding. Software engineering is difficult. Programs of any substantial length are inherently prone to errors of all kinds. The development cycle is full of traps unknown to the apprentice developer. Yet, in Python textbooks little attention is paid to this aspect of getting your code to run. At most, there is a chapter on debugging or unit testing in your average basic Python book. However, the proportion of time spent on getting your code to run is much higher in the real world. Pro Python Best Practices aims to solve this problem. What You'll Learn Learn common debugging techniques that help you find and eliminate errors Gain techniques to detect bugs more easily discover best="" practices="" to="" prevent="" bugscarry="" out="" automated="" testing="" discover="" problems="" fasteruse="" maintain="" a="" project="" over="" long="" timeLearn techniques to keep your project under controlbr/uldivbWho This Book Is For/bbr/divdivbr/divdivExperienced Python coders from web development, big data, and more./divdivbr/divdivdiv/div Software Project Survival Guide *Microsoft Press* Equip yourself with SOFTWARE PROJECT SURVIVAL GUIDE. It's for everyone with a stake in the outcome of a development project--and especially for those without formal software project management training. That includes top managers, executives, clients, investors, end-user representatives, project managers, and technical leads. Here you'll find guidance from the acclaimed author of the classics CODE COMPLETE and RAPID DEVELOPMENT. Steve McConnell draws on solid research and a career's worth of hard-won experience to map the surest path to your goal--what he calls "one specific approach to software development that works pretty well most of the time for most projects." Nineteen chapters in four sections cover the concepts and strategies you need for mastering the development process, including planning, design, management, quality assurance, testing, and archiving. For newcomers and seasoned project managers alike, SOFTWARE PROJECT SURVIVAL GUIDE draws on a vast store of techniques to create an elegantly simplified and reliable framework for project management success. So don't worry about wandering among complex sets of project management techniques that require years to sort out and master. SOFTWARE PROJECT SURVIVAL GUIDE goes straight to the heart of the matter to help your projects succeed. And that makes it a required addition to every professional's bookshelf. Information Systems: Development, Research, Applications, Education 9th SIGSAND/PLAIS EuroSymposium 2016, Gdansk, Poland, September 29, 2016, Proceedings *Springer* This book constitutes the refereed proceedings of the SIGSAND/PLAIS EuroSymposium 2016 titled Information Systems: Development, Research, Applications, Education, held in Gdansk and Sopot, Poland, on September 29, 2016. The objective of this symposium is to promote and develop high-quality research on all issues related to systems analysis and design (SAND). It provides a forum for SAND researchers and practitioners in Europe and beyond to interact, collaborate, and develop their field. The 14 papers presented in this volume were carefully reviewed and selected from 34 submissions. They are organized in topical sections on information systems development, information systems management, and information systems learning. Modern PHP New Features and Good Practices *"O'Reilly Media, Inc."* PHP is experiencing a renaissance, though it may be difficult to tell with all of the outdated PHP tutorials online. With this practical guide, you'll learn how PHP has become a full-featured, mature language with object-orientation, namespaces, and a growing collection of reusable component libraries. Author Josh Lockhart—creator of PHP The Right Way, a popular initiative to encourage PHP best practices—reveals these new language features in action. You'll learn best practices for application architecture and planning, databases, security, testing, debugging, and deployment. If you have a basic understanding of PHP and want to bolster your skills, this is your book. Learn modern PHP features, such as namespaces, traits, generators, and closures Discover how to find, use, and create PHP components Follow best practices for application security, working with databases, errors and exceptions, and more Learn tools and techniques for deploying, tuning, testing, and profiling your PHP applications Explore Facebook's HVVM and Hack language implementations—and how they affect modern PHP Build a local development environment that closely matches your production server Rapid Application Development *Macmillan Coll Division* Optimized C++ Proven Techniques for Heightened Performance *"O'Reilly Media, Inc."* In today's fast and competitive world, a program's performance is just as important to customers as the features it provides. This practical guide teaches developers performance-tuning principles that enable optimization in C++. You'll learn how to make code that already embodies best practices of C++ design run faster and consume fewer resources on any computer--whether it's a watch, phone, workstation, supercomputer, or globe-spanning network of servers. Author Kurt Guntheroth provides several running examples that demonstrate how to apply these principles incrementally to improve existing code so it meets customer requirements for responsiveness and throughput. The advice in this book will prove itself the first time you hear a colleague exclaim, "Wow, that was fast. Who fixed something?"Locate performance hot spots using the profiler and software timersLearn to perform repeatable experiments to measure performance of code changesOptimize use of dynamically allocated variablesImprove performance of hot loops and functionsSpeed up string handling functionsRecognize efficient algorithms and optimization patternsLearn the strengths--and weaknesses--of C++ container classesView searching and sorting through an optimizer's eyeMake efficient use of C++ streaming I/O functionsUse C++ thread-based concurrency features effectively Deep Learning for Coders with fastai and PyTorch *"O'Reilly Media, Inc."* Deep learning is often viewed as the exclusive domain of math PhDs and big tech companies. But as this hands-on guide demonstrates, programmers comfortable with Python can achieve impressive results in deep learning with little math background, small amounts of data, and minimal code. How? With fastai, the first library to provide a consistent interface to the most frequently used deep learning applications. Authors Jeremy

Howard and Sylvain Gugger, the creators of fastai, show you how to train a model on a wide range of tasks using fastai and PyTorch. You'll also dive progressively further into deep learning theory to gain a complete understanding of the algorithms behind the scenes. Train models in computer vision, natural language processing, tabular data, and collaborative filtering Learn the latest deep learning techniques that matter most in practice Improve accuracy, speed, and reliability by understanding how deep learning models work Discover how to turn your models into web applications Implement deep learning algorithms from scratch Consider the ethical implications of your work Gain insight from the foreword by PyTorch cofounder, Soumith Chintala Infosec Strategies and Best Practices Gain proficiency in information security using expert-level strategies and best practices *Packt Publishing Ltd* Advance your career as an information security professional by turning theory into robust solutions to secure your organization Key FeaturesConvert the theory of your security certifications into actionable changes to secure your organizationDiscover how to structure policies and procedures in order to operationalize your organization's information security strategyLearn how to achieve security goals in your organization and reduce software riskBook Description Information security and risk management best practices enable professionals to plan, implement, measure, and test their organization's systems and ensure that they're adequately protected against threats. The book starts by helping you to understand the core principles of information security, why risk management is important, and how you can drive information security governance. You'll then explore methods for implementing security controls to achieve the organization's information security goals. As you make progress, you'll get to grips with design principles that can be utilized along with methods to assess and mitigate architectural vulnerabilities. The book will also help you to discover best practices for designing secure network architectures and controlling and managing third-party identity services. Finally, you will learn about designing and managing security testing processes, along with ways in which you can improve software security. By the end of this infosec book, you'll have learned how to make your organization less vulnerable to threats and reduce the likelihood and impact of exploitation. As a result, you will be able to make an impactful change in your organization toward a higher level of information security. What you will learnUnderstand and operationalize risk management concepts and important security operations activitiesDiscover how to identify, classify, and maintain information and assetsAssess and mitigate vulnerabilities in information systemsDetermine how security control testing will be undertakenIncorporate security into the SDLC (software development life cycle)Improve the security of developed software and mitigate the risks of using unsafe softwareWho this book is for If you are looking to begin your career in an information security role, then this book is for you. Anyone who is studying to achieve industry-standard certification such as the CISSP or CISM, but looking for a way to convert concepts (and the seemingly endless number of acronyms) from theory into practice and start making a difference in your day-to-day work will find this book useful. IBM WebSphere Application Server V8 Concepts, Planning, and Design Guide *IBM Redbooks* This IBM® Redbooks® publication provides information about the concepts, planning, and design of IBM WebSphere® Application Server V8 environments. The target audience of this book is IT architects and consultants who want more information about the planning and designing of application-serving environments, from small to large, and complex implementations. This book addresses the packaging and features in WebSphere Application Server V8 and highlights the most common implementation topologies. It provides information about planning for specific tasks and components that conform to the WebSphere Application Server environment. Also in this book are planning guidelines for WebSphere Application Server V8 and WebSphere Application Server Network Deployment V8 on distributed platforms and for WebSphere Application Server for z/OS® V8. This book contains information about migration considerations when moving from previous releases. Modern Programming: Object Oriented Programming and Best Practices Deconstruct object-oriented programming and use it with other programming paradigms to build applications *Packt Publishing Ltd* Discover the untapped features of object-oriented programming and use it with other software tools to code fast, efficient applications. Key FeaturesExplore the complexities of object-oriented programming (OOP)Discover what OOP can do for youLearn to use the key tools and software engineering practices to support your own programming needsBook Description Your experience and knowledge always influence the approach you take and the tools you use to write your programs. With a sound understanding of how to approach your goal and what software paradigms to use, you can create high-performing applications quickly and efficiently. In this two-part book, you'll discover the untapped features of object-oriented programming and use it with other software tools to code fast and efficient applications. The first part of the book begins with a discussion on how OOP is used today and moves on to analyze the ideas and problems that OOP doesn't address. It continues by deconstructing the complexity of OOP, showing you its fundamentally simple core. You'll see that, by using the distinctive elements of OOP, you can learn to build your applications more easily. The next part of this book talks about acquiring the skills to become a better programmer. You'll get an overview of how various tools, such as version control and build management, help make your life easier. This book also discusses the pros and cons of other programming paradigms, such as aspect-oriented programming and functional programming, and helps to select the correct approach for your projects. It ends by talking about the philosophy behind designing software and what it means to be a "good" developer. By the end of this two-part book, you will have learned that OOP is not always complex, and you will know how you can evolve into a better programmer by learning about ethics, teamwork, and documentation. What you will learnUntangle the complexity of object-oriented programming by breaking it down to its essential building blocksRealize the full potential of OOP to design efficient, maintainable programsUtilize coding best practices, including TDD, pair programming and code reviews, to improve your workUse tools, such as source control and IDEs, to work more efficientlyLearn how to most productively work with other developersBuild your own software development philosophyWho this book is for This book is ideal for programmers who want to understand the philosophy behind creating software and what it means to be "good" at

designing software. Programmers who want to deconstruct the OOP paradigm and see how it can be reconstructed in a clear, straightforward way will also find this book useful. To understand the ideas expressed in this book, you must be an experienced programmer who wants to evolve their practice. Lotus Notes Developer's Toolbox Tips for Rapid and Successful Deployment *IBM Press* "Lotus Notes Developer's Toolbox" focuses on the core issues, interfaces, quirks, and tips for application development. It provides everything needed to streamline the design, development, and support of Lotus Notes applications. Unlike any other publication on the market, this book approaches the learning process from the perspective of an experienced developer. Occupational Outlook Handbook Oracle PL/SQL Best Practices Optimizing Oracle Code *"O'Reilly Media, Inc."* **In this book, Steven Feuerstein, widely recognized as one of the world's experts on the Oracle PL/SQL language, distills his many years of programming, writing, and teaching about PL/SQL into a set of PL/SQL language "best practices"--rules for writing code that is readable, maintainable, and efficient. Too often, developers focus on simply writing programs that run without errors--and ignore the impact of poorly written code upon both system performance and their ability (and their colleagues' ability) to maintain that code over time.Oracle PL/SQL Best Practices is a concise, easy-to-use reference to Feuerstein's recommendations for excellent PL/SQL coding. It answers the kinds of questions PL/SQL developers most frequently ask about their code: How should I format my code? What naming conventions, if any, should I use? How can I write my packages so they can be more easily maintained? What is the most efficient way to query information from the database? How can I get all the developers on my team to handle errors the same way? The book contains 120 best practices, divided by topic area. It's full of advice on the program development process, coding style, writing SQL in PL/SQL, data structures, control structures, exception handling, program and package construction, and built-in packages. It also contains a handy, pull-out quick reference card. As a helpful supplement to the text, code examples demonstrating each of the best practices are available on the O'Reilly web site.Oracle PL/SQL Best Practices is intended as a companion to O'Reilly's larger Oracle PL/SQL books. It's a compact, readable reference that you'll turn to again and again--a book that no serious developer can afford to be without. Accelerate The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations** *IT Revolution* **Winner of the Shingo Publication Award Accelerate your organization to win in the marketplace. How can we apply technology to drive business value? For years, we've been told that the performance of software delivery teams doesn't matter—that it can't provide a competitive advantage to our companies. Through four years of groundbreaking research to include data collected from the State of DevOps reports conducted with Puppet, Dr. Nicole Forsgren, Jez Humble, and Gene Kim set out to find a way to measure software delivery performance—and what drives it—using rigorous statistical methods. This book presents both the findings and the science behind that research, making the information accessible for readers to apply in their own organizations. Readers will discover how to measure the performance of their teams, and what capabilities they should invest in to drive higher performance. This book is ideal for management at every level. More Effective Agile A Roadmap for Software Leaders In this comprehensive yet accessible overview for software leaders, the author presents an impactful, action-oriented prescription-covering the practical considerations needed to ensure you reap the full benefits of effective Agile**