

Head First Design Patterns

Clean Code
Head First JavaScript Programming
Head First JavaScript Object-Oriented Design And Patterns
Learning JavaScript Design Patterns
Game Programming Patterns
Hands-On Design Patterns with Java
Head First Design Patterns, 2nd Edition
SCJP Sun Certified Programmer for Java 6 Study Guide
Head First Learn to Code
Effective Java
Head First Servlets and JSP
Design Patterns Explained
Head First HTML and CSS
The Natural Navigator
Head First Design Patterns
Software Engineering at Google
The Pragmatic Programmer
Head First Python
Story-Based Inquiry: A Manual for Investigative Journalists
C# 3.0 Design Patterns
Head First HTML with CSS & XHTML
Head First HTML5 Programming
Codeless Data Structures and Algorithms
Node.js Design Patterns
Head First C#
Java Design Patterns
You are Not So Smart
Head First EJB
Design Patterns
Java Generics and Collections
Design Patterns For Dummies
Head First Software Development
Head First Object-Oriented Analysis and Design
Clean Agile
Design Patterns in Ruby (Adobe Reader)
Head First Java
Understanding Machine Learning
Effective DevOps
Design Patterns in C#

Clean Code

Some companies think that adopting devops means bringing in specialists or a host of new tools. With this practical guide, you'll learn why devops is a professional and cultural movement that calls for change from inside your organization. Authors Ryn Daniels and Jennifer Davis provide several approaches for improving collaboration within teams, creating affinity among teams, promoting efficient tool usage in your company, and scaling up what works throughout your organization's inflection points. Devops stresses iterative efforts to break down information silos, monitor relationships, and repair misunderstandings that arise between and within teams in your organization. By applying the actionable strategies in this book, you can make sustainable changes in your environment regardless of your level within your organization. Explore the foundations of devops and learn the four pillars of effective devops

- Encourage collaboration to help individuals work together and build durable and long-lasting relationships
- Create affinity among teams while balancing differing goals or metrics
- Accelerate cultural direction by selecting tools and workflows that complement your organization
- Troubleshoot common problems and misunderstandings that can arise throughout the organizational lifecycle

Learn from case studies from organizations and individuals to help inform your own devops journey

Head First JavaScript Programming

In the era of self-taught developers and programmers, essential topics in the industry are frequently learned without a formal academic foundation. A solid grasp of data structures and algorithms (DSA) is imperative for anyone looking to do professional software development and engineering, but classes in the subject can be dry or spend too much time on theory and unnecessary readings. Regardless of your programming language background, Codeless Data Structures and Algorithms has you covered. In this book, author Armstrong Subero will help you learn DSAs without writing a single line of code. Straightforward explanations and diagrams give you a confident handle on the topic while ensuring you never

have to open your code editor, use a compiler, or look at an integrated development environment. Subero introduces you to linear, tree, and hash data structures and gives you important insights behind the most common algorithms that you can directly apply to your own programs. Codeless Data Structures and Algorithms provides you with the knowledge about DSAs that you will need in the professional programming world, without using any complex mathematics or irrelevant information. Whether you are a new developer seeking a basic understanding of the subject or a decision-maker wanting a grasp of algorithms to apply to your projects, this book belongs on your shelf. Quite often, a new, refreshing, and unpretentious approach to a topic is all you need to get inspired. What You'll Learn Understand tree data structures without delving into unnecessary details or going into too much theory Get started learning linear data structures with a basic discussion on computer memory Study an overview of arrays, linked lists, stacks and queues Who This Book Is For This book is for beginners, self-taught developers and programmers, and anyone who wants to understand data structures and algorithms but don't want to wade through unnecessary details about quirks of a programming language or don't have time to sit and read a massive book on the subject. This book is also useful for non-technical decision-makers who are curious about how algorithms work.

Head First JavaScript

Introduces machine learning and its algorithmic paradigms, explaining the principles behind automated learning approaches and the considerations underlying their usage.

Object-Oriented Design And Patterns

A guide to the programming language describes how to build Python-enabled Web servers and applications, write mobile apps on the Android platform, develop sophisticated games, build GUI-based programs, and write Python scripts to automate tasks.

Learning JavaScript Design Patterns

Capturing a wealth of experience about the design of object-oriented software, four top-notch designers present a catalog of simple and succinct solutions to commonly occurring design problems. Previously undocumented, these 23 patterns allow designers to create more flexible, elegant, and ultimately reusable designs without having to rediscover the design solutions themselves. The authors begin by describing what patterns are and how they can help you design object-oriented software. They then go on to systematically name, explain, evaluate, and catalog recurring designs in object-oriented systems. With Design Patterns as your guide, you will learn how these important patterns fit into the software development process, and how you can leverage them to solve your own design problems most efficiently. Each pattern describes the circumstances in which it is applicable, when it can be applied in view of other design constraints, and the consequences and trade-offs of using the pattern within a larger design. All patterns are compiled from real systems and are based on real-world examples.

Each pattern also includes code that demonstrates how it may be implemented in object-oriented programming languages like C++ or Smalltalk.

Game Programming Patterns

Are you looking for a deeper understanding of the Java™ programming language so that you can write code that is clearer, more correct, more robust, and more reusable? Look no further! *Effective Java™, Second Edition*, brings together seventy-eight indispensable programmer's rules of thumb: working, best-practice solutions for the programming challenges you encounter every day. This highly anticipated new edition of the classic, Jolt Award-winning work has been thoroughly updated to cover Java SE 5 and Java SE 6 features introduced since the first edition. Bloch explores new design patterns and language idioms, showing you how to make the most of features ranging from generics to enums, annotations to autoboxing. Each chapter in the book consists of several "items" presented in the form of a short, standalone essay that provides specific advice, insight into Java platform subtleties, and outstanding code examples. The comprehensive descriptions and explanations for each item illuminate what to do, what not to do, and why. Highlights include: New coverage of generics, enums, annotations, autoboxing, the for-each loop, varargs, concurrency utilities, and much more Updated techniques and best practices on classic topics, including objects, classes, libraries, methods, and serialization How to avoid the traps and pitfalls of commonly misunderstood subtleties of the language Focus on the language and its most fundamental libraries: `java.lang`, `java.util`, and, to a lesser extent, `java.util.concurrent` and `java.io` Simply put, *Effective Java™, Second Edition*, presents the most practical, authoritative guidelines available for writing efficient, well-designed programs.

Hands-On Design Patterns with Java

Praise for *Design Patterns in Ruby* " *Design Patterns in Ruby* documents smart ways to resolve many problems that Ruby developers commonly encounter. Russ Olsen has done a great job of selecting classic patterns and augmenting these with newer patterns that have special relevance for Ruby. He clearly explains each idea, making a wealth of experience available to Ruby developers for their own daily work." —Steve Metsker, Managing Consultant with Dominion Digital, Inc. "This book provides a great demonstration of the key 'Gang of Four' design patterns without resorting to overly technical explanations. Written in a precise, yet almost informal style, this book covers enough ground that even those without prior exposure to design patterns will soon feel confident applying them using Ruby. Olsen has done a great job to make a book about a classically 'dry' subject into such an engaging and even occasionally humorous read." —Peter Cooper "This book renewed my interest in understanding patterns after a decade of good intentions. Russ picked the most useful patterns for Ruby and introduced them in a straightforward and logical manner, going beyond the GoF's patterns. This book has improved my use of Ruby, and encouraged me to blow off the dust covering the GoF book." —Mike Stok " *Design Patterns in Ruby* is a great way for programmers from statically typed objectoriented languages to learn how design patterns appear in a more dynamic, flexible language like Ruby." —Rob Sanheim, *Ruby Ninja*, *Relevance* Most design pattern books are based on C++ and Java. But

Ruby is different—and the language's unique qualities make design patterns easier to implement and use. In this book, Russ Olsen demonstrates how to combine Ruby's power and elegance with patterns, and write more sophisticated, effective software with far fewer lines of code. After reviewing the history, concepts, and goals of design patterns, Olsen offers a quick tour of the Ruby language—enough to allow any experienced software developer to immediately utilize patterns with Ruby. The book especially calls attention to Ruby features that simplify the use of patterns, including dynamic typing, code closures, and "mixins" for easier code reuse. Fourteen of the classic "Gang of Four" patterns are considered from the Ruby point of view, explaining what problems each pattern solves, discussing whether traditional implementations make sense in the Ruby environment, and introducing Ruby-specific improvements. You'll discover opportunities to implement patterns in just one or two lines of code, instead of the endlessly repeated boilerplate that conventional languages often require. Design Patterns in Ruby also identifies innovative new patterns that have emerged from the Ruby community. These include ways to create custom objects with metaprogramming, as well as the ambitious Rails-based "Convention Over Configuration" pattern, designed to help integrate entire applications and frameworks. Engaging, practical, and accessible, Design Patterns in Ruby will help you build better software while making your Ruby programming experience more rewarding.

Head First Design Patterns, 2nd Edition

Get hands-on experience implementing 26 of the most common design patterns using Java and Eclipse. In addition to Gang of Four (GoF) design patterns, you will also learn about alternative design patterns, and understand the criticisms of design patterns with an overview of anti-patterns. For each pattern you will see at least one real-world scenario, a computer-world example, and a complete implementation including output. This book has three parts. The first part covers 23 Gang of Four (GoF) design patterns. The second part includes three alternative design patterns. The third part presents criticisms of design patterns with an overview of anti-patterns. You will work through easy-to-follow examples to understand the concepts in depth and you will have a collection of programs to port over to your own projects. A Q&A session is included in each chapter and covers the pros and cons of each pattern. The last chapter presents FAQs about the design patterns. The step-by-step approach of the book helps you apply your skills to learn other patterns on your own, and to be familiar with the latest version of Java and Eclipse. What You'll Learn Work with each of the design patterns Implement design patterns in real-world applications Choose from alternative design patterns by comparing their pros and cons Use the Eclipse IDE to write code and generate output Read the in-depth Q&A session in each chapter with pros and cons for each design pattern Who This Book Is For Software developers, architects, and programmers

SCJP Sun Certified Programmer for Java 6 Study Guide

With Learning JavaScript Design Patterns, you'll learn how to write beautiful, structured, and maintainable JavaScript by applying classical and modern design patterns to the language. If you want to keep your code efficient, more manageable, and up-to-date with the latest best practices, this book is for you.

Explore many popular design patterns, including Modules, Observers, Facades, and Mediators. Learn how modern architectural patterns—such as MVC, MVP, and MVVM—are useful from the perspective of a modern web application developer. This book also walks experienced JavaScript developers through modern module formats, how to namespace code effectively, and other essential topics. Learn the structure of design patterns and how they are written Understand different pattern categories, including creational, structural, and behavioral Walk through more than 20 classical and modern design patterns in JavaScript Use several options for writing modular code—including the Module pattern, Asynchronous Module Definition (AMD), and CommonJS Discover design patterns implemented in the jQuery library Learn popular design patterns for writing maintainable jQuery plugins "This book should be in every JavaScript developer's hands. It's the go-to book on JavaScript patterns that will be read and referenced many times in the future."—Andrée Hansson, Lead Front-End Developer, [presis!](#)

Head First Learn to Code

Using research in neurobiology, cognitive science and learning theory, this text loads patterns into your brain in a way that lets you put them to work immediately, makes you better at solving software design problems, and improves your ability to speak the language of patterns with others on your team.

Effective Java

Presents information on using HTML5 to create dynamic, data-rich Web pages, covering such topics as geolocation, 2D drawing, Web storages, and Web workers.

Head First Servlets and JSP

If you want to speed up the development of your .NET applications, you're ready for C# design patterns -- elegant, accepted and proven ways to tackle common programming problems. This practical guide offers you a clear introduction to the classic object-oriented design patterns, and explains how to use the latest features of C# 3.0 to code them. C# Design Patterns draws on new C# 3.0 language and .NET 3.5 framework features to implement the 23 foundational patterns known to working developers. You get plenty of case studies that reveal how each pattern is used in practice, and an insightful comparison of patterns and where they would be best used or combined. This well-organized and illustrated book includes: An explanation of design patterns and why they're used, with tables and guidelines to help you choose one pattern over another Illustrated coverage of each classic Creational, Structural, and Behavioral design pattern, including its representation in UML and the roles of its various players C# 3.0 features introduced by example and summarized in sidebars for easy reference Examples of each pattern at work in a real .NET 3.5 program available for download from O'Reilly and the author's companion web site Quizzes and exercises to test your understanding of the material. With C# 3.0 Design Patterns, you learn to make code correct, extensible and efficient to save time up front and eliminate problems later. If your business relies on efficient application development and quality code, you need C# Design Patterns.

Design Patterns Explained

Provides information on successful software development, covering such topics as customer requirements, task estimates, principles of good design, dealing with source code, system testing, and handling bugs.

Head First HTML and CSS

The biggest challenge facing many game programmers is completing their game. Most game projects fizzle out, overwhelmed by the complexity of their own code. Game Programming Patterns tackles that exact problem. Based on years of experience in shipped AAA titles, this book collects proven patterns to untangle and optimize your game, organized as independent recipes so you can pick just the patterns you need. You will learn how to write a robust game loop, how to organize your entities using components, and take advantage of the CPUs cache to improve your performance. You'll dive deep into how scripting engines encode behavior, how quadrees and other spatial partitions optimize your engine, and how other classic design patterns can be used in games.

The Natural Navigator

Provides information on analyzing, designing, and writing object-oriented software.

Head First Design Patterns

Software Engineering at Google

Presents information on creating Web pages using HTML and CSS.

The Pragmatic Programmer

A guide to JavaBeans provides more than two hundred questions and answers to help readers pass the Sun Certified Business Component Developer exam.

Head First Python

Learning a complex new language is no easy task especially when it s an object-oriented computer programming language like Java. You might think the problem is your brain. It seems to have a mind of its own, a mind that doesn't always want to take in the dry, technical stuff you're forced to study. The fact is your brain craves novelty. It's constantly searching, scanning, waiting for something unusual to happen. After all, that's the way it was built to help you stay alive. It takes all the routine, ordinary, dull stuff and filters it to the background so it won't interfere with your brain's real work--recording things that matter. How does your brain know what matters? It's like the creators of the Head First approach say, suppose you're out for a hike and a tiger jumps in front of you, what happens in your brain? Neurons fire. Emotions crank up. Chemicals surge. That's how your brain knows. And that's how your brain will learn Java. Head First Java combines puzzles, strong

visuals, mysteries, and soul-searching interviews with famous Java objects to engage you in many different ways. It's fast, it's fun, and it's effective. And, despite its playful appearance, Head First Java is serious stuff: a complete introduction to object-oriented programming and Java. You'll learn everything from the fundamentals to advanced topics, including threads, network sockets, and distributed programming with RMI. And the new, second edition focuses on Java 5.0, the latest version of the Java language and development platform. Because Java 5.0 is a major update to the platform, with deep, code-level changes, even more careful study and implementation is required. So learning the Head First way is more important than ever. If you've read a Head First book, you know what to expect--a visually rich format designed for the way your brain works. If you haven't, you're in for a treat. You'll see why people say it's unlike any other Java book you've ever read. By exploiting how your brain works, Head First Java compresses the time it takes to learn and retain--complex information. Its unique approach not only shows you what you need to know about Java syntax, it teaches you to think like a Java programmer. If you want to be bored, buy some other book. But if you want to understand Java, this book's for you.

Story-Based Inquiry: A Manual for Investigative Journalists

There's a pattern here, and here's how to use it! Find out how the 23 leading design patterns can save you time and trouble Ever feel as if you've solved this programming problem before? You -- or someone -- probably did, and that's why there's a design pattern to help this time around. This book shows you how (and when) to use the famous patterns developed by the "Gang of Four," plus some new ones, all designed to make your programming life easier. Discover how to: * Simplify the programming process with design patterns * Make the most of the Decorator, Factory, and Adapter patterns * Identify which pattern applies * Reduce the amount of code needed for a task * Create your own patterns

C# 3.0 Design Patterns

What will you learn from this book? It's no secret the world around you is becoming more connected, more configurable, more programmable, more computational. You can remain a passive participant, or you can learn to code. With Head First Learn to Code you'll learn how to think computationally and how to write code to make your computer, mobile device, or anything with a CPU do things for you. Using the Python programming language, you'll learn step by step the core concepts of programming as well as many fundamental topics from computer science, such as data structures, storage, abstraction, recursion, and modularity. Why does this book look so different? Based on the latest research in cognitive science and learning theory, Head First Learn to Code uses a visually rich format to engage your mind, rather than a text-heavy approach that puts you to sleep. Why waste your time struggling with new concepts? This multi-sensory learning experience is designed for the way your brain really works.

Head First HTML with CSS & XHTML

Looks at the principles and clean code, includes case studies showcasing the

practices of writing clean code, and contains a list of heuristics and "smells" accumulated from the process of writing clean code.

Head First HTML5 Programming

"One of the most significant books in my life." –Obie Fernandez, Author, The Rails Way
"Twenty years ago, the first edition of The Pragmatic Programmer completely changed the trajectory of my career. This new edition could do the same for yours." –Mike Cohn, Author of Succeeding with Agile, Agile Estimating and Planning, and User Stories Applied
". . . filled with practical advice, both technical and professional, that will serve you and your projects well for years to come."
–Andrea Goulet, CEO, Corgibytes, Founder, LegacyCode.Rocks
". . . lightning does strike twice, and this book is proof."
–VM (Vicky) Brasseur, Director of Open Source Strategy, Juniper Networks
The Pragmatic Programmer is one of those rare tech books you'll read, re-read, and read again over the years. Whether you're new to the field or an experienced practitioner, you'll come away with fresh insights each and every time. Dave Thomas and Andy Hunt wrote the first edition of this influential book in 1999 to help their clients create better software and rediscover the joy of coding. These lessons have helped a generation of programmers examine the very essence of software development, independent of any particular language, framework, or methodology, and the Pragmatic philosophy has spawned hundreds of books, screencasts, and audio books, as well as thousands of careers and success stories. Now, twenty years later, this new edition re-examines what it means to be a modern programmer. Topics range 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
- Learn continuously
- Avoid the trap of duplicating knowledge
- Write flexible, dynamic, and adaptable code
- Harness the power of basic tools
- Avoid programming by coincidence
- Learn real requirements
- Solve the underlying problems of concurrent code
- Guard against security vulnerabilities
- Build teams of Pragmatic Programmers
- Take responsibility for your work and career
- Test ruthlessly and effectively, including property-based testing
- Implement the Pragmatic Starter Kit
- Delight your users

Written as a series of self-contained sections and filled with classic and fresh anecdotes, thoughtful examples, and interesting analogies, The Pragmatic Programmer illustrates the best approaches 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. Register your book for convenient access to downloads, updates, and/or corrections as they become available. See inside book for details.

Codeless Data Structures and Algorithms

"A complete learning experience for creating industry standard Web pages - but you won't be just reading: you'll be playing games, solving puzzles, pondering mysteries and creating Web pages like you never imagined. You'll be also learning how HTML works with CSS . . . if you're going to create Web pages in the 21st century, then you want to know and to understand CSS, too."

Node.js Design Patterns

A guide to C# 3.0 and Visual Studio 2008 covers such topics as objects, data types and references, encapsulation, interfaces, exception handling, and LINQ.

Head First C#

The Best Fully Integrated Study System Available--Written by the Lead Developers of Exam 310-065 With hundreds of practice questions and hands-on exercises, SCJP Sun Certified Programmer for Java 6 Study Guide covers what you need to know--and shows you how to prepare--for this challenging exam. 100% complete coverage of all official objectives for exam 310-065 Exam Objective Highlights in every chapter point out certification objectives to ensure you're focused on passing the exam Exam Watch sections in every chapter highlight key exam topics covered Simulated exam questions match the format, tone, topics, and difficulty of the real exam Covers all SCJP exam topics, including: Declarations and Access Control · Object Orientation · Assignments · Operators · Flow Control, Exceptions, and Assertions · Strings, I/O, Formatting, and Parsing · Generics and Collections · Inner Classes · Threads · Development CD-ROM includes: Complete MasterExam practice testing engine, featuring: Two full practice exams; Detailed answers with explanations; Score Report performance assessment tool Electronic book for studying on the go Bonus coverage of the SCJD exam included! Bonus downloadable MasterExam practice test with free online registration.

Java Design Patterns

So you're ready to make the leap from writing HTML and CSS web pages to creating dynamic web applications. You want to take your web skills to the next level. And you're finally ready to add "programmer" to the resume. It sounds like you're ready to learn the Web's hottest programming language: JavaScript. Head First JavaScript is your ticket to going beyond copying and pasting the code from someone else's web site, and writing your own interactive web pages. With Head First JavaScript, you learn: The basics of programming, from variables to types to looping How the web browser runs your code, and how you can talk to the browser with your code Why you'll never have to worry about casting, overloading, or polymorphism when you're writing JavaScript code How to use the Document Object Model to change your web pages without making your users click buttons If you've ever read a Head First book, you know what to expect -- a visually rich format designed for the way your brain works. Head First JavaScript is no exception. It starts where HTML and CSS leave off, and takes you through your first program into more complex programming concepts -- like working directly with the web browser's object model and writing code that works on all modern browsers. Don't be intimidated if you've never written a line of code before! In typical Head First style, Head First JavaScript doesn't skip steps, and we're not interested in having you cut and paste code. You'll learn JavaScript, understand it, and have a blast along the way. So get ready dynamic and exciting web pages are just pages away.

You are Not So Smart

Explains how self-delusion is part of a person's psychological defense system, identifying common misconceptions people have on topics such as caffeine withdrawal, hindsight, and brand loyalty.

Head First EJB

Get hands-on experience with each Gang of Four design pattern using C#. For each of the patterns, you'll see at least one real-world scenario, a coding example, and a complete implementation including output. In the first part of Design Patterns in C#, you will cover the 23 Gang of Four (GoF) design patterns, before moving onto some alternative design patterns, including the Simple Factory Pattern, the Null Object Pattern, and the MVC Pattern. The final part winds up with a conclusion and criticisms of design patterns with chapters on anti-patterns and memory leaks. By working through easy-to-follow examples, you will understand the concepts in depth and have a collection of programs to port over to your own projects. Along the way, the author discusses the different creational, structural, and behavioral patterns and why such classifications are useful. In each of these chapters, there is a Q&A session that clears up any doubts and covers the pros and cons of each of these patterns. He finishes the book with FAQs that will help you consolidate your knowledge. This book presents the topic of design patterns in C# in such a way that anyone can grasp the idea. What You Will Learn Work with each of the design patterns Implement the design patterns in real-world applications Select an alternative to these patterns by comparing their pros and cons Use Visual Studio Community Edition 2017 to write code and generate output Who This Book Is For Software developers, software testers, and software architects.

Design Patterns

"One of the great things about the book is the way the authors explain concepts very simply using analogies rather than programming examples—this has been very inspiring for a product I'm working on: an audio-only introduction to OOP and software development." —Bruce Eckel "I would expect that readers with a basic understanding of object-oriented programming and design would find this book useful, before approaching design patterns completely. Design Patterns Explained complements the existing design patterns texts and may perform a very useful role, fitting between introductory texts such as UML Distilled and the more advanced patterns books." —James Noble Leverage the quality and productivity benefits of patterns—without the complexity! Design Patterns Explained, Second Edition is the field's simplest, clearest, most practical introduction to patterns. Using dozens of updated Java examples, it shows programmers and architects exactly how to use patterns to design, develop, and deliver software far more effectively. You'll start with a complete overview of the fundamental principles of patterns, and the role of object-oriented analysis and design in contemporary software development. Then, using easy-to-understand sample code, Alan Shalloway and James Trott illuminate dozens of today's most useful patterns: their underlying concepts, advantages, tradeoffs, implementation techniques, and pitfalls to avoid. Many patterns are accompanied by UML diagrams. Building on their best-selling First Edition, Shalloway and Trott have thoroughly updated this book to reflect new software design trends, patterns, and implementation techniques. Reflecting extensive reader feedback, they have deepened and

clarified coverage throughout, and reorganized content for even greater ease of understanding. New and revamped coverage in this edition includes Better ways to start "thinking in patterns" How design patterns can facilitate agile development using eXtreme Programming and other methods How to use commonality and variability analysis to design application architectures The key role of testing into a patterns-driven development process How to use factories to instantiate and manage objects more effectively The Object-Pool Pattern—a new pattern not identified by the "Gang of Four" New study/practice questions at the end of every chapter Gentle yet thorough, this book assumes no patterns experience whatsoever. It's the ideal "first book" on patterns, and a perfect complement to Gamma's classic Design Patterns. If you're a programmer or architect who wants the clearest possible understanding of design patterns—or if you've struggled to make them work for you—read this book.

Java Generics and Collections

Agile Values and Principles for a New Generation “In the journey to all things Agile, Uncle Bob has been there, done that, and has the both the t-shirt and the scars to show for it. This delightful book is part history, part personal stories, and all wisdom. If you want to understand what Agile is and how it came to be, this is the book for you.” –Grady Booch “Bob’s frustration colors every sentence of Clean Agile, but it’s a justified frustration. What is in the world of Agile development is nothing compared to what could be. This book is Bob’s perspective on what to focus on to get to that ‘what could be.’ And he’s been there, so it’s worth listening.” –Kent Beck “It’s good to read Uncle Bob’s take on Agile. Whether just beginning, or a seasoned Agilista, you would do well to read this book. I agree with almost all of it. It’s just some of the parts make me realize my own shortcomings, dammit. It made me double-check our code coverage (85.09%).” –Jon Kern Nearly twenty years after the Agile Manifesto was first presented, the legendary Robert C. Martin (“Uncle Bob”) reintroduces Agile values and principles for a new generation—programmers and nonprogrammers alike. Martin, author of Clean Code and other highly influential software development guides, was there at Agile’s founding. Now, in Clean Agile: Back to Basics, he strips away misunderstandings and distractions that over the years have made it harder to use Agile than was originally intended. Martin describes what Agile is in no uncertain terms: a small discipline that helps small teams manage small projects . . . with huge implications because every big project is comprised of many small projects. Drawing on his fifty years’ experience with projects of every conceivable type, he shows how Agile can help you bring true professionalism to software development. Get back to the basics—what Agile is, was, and should always be Understand the origins, and proper practice, of SCRUM Master essential business-facing Agile practices, from small releases and acceptance tests to whole-team communication Explore Agile team members’ relationships with each other, and with their product Rediscover indispensable Agile technical practices: TDD, refactoring, simple design, and pair programming Understand the central roles values and craftsmanship play in your Agile team’s success If you want Agile’s true benefits, there are no shortcuts: You need to do Agile right. Clean Agile: Back to Basics will show you how, whether you’re a developer, tester, manager, project manager, or customer. Register your book for convenient access to downloads, updates, and/or corrections as they become available. See inside book for details.

Design Patterns For Dummies

Cay Horstmann offers readers an effective means for mastering computing concepts and developing strong design skills. This book introduces object-oriented fundamentals critical to designing software and shows how to implement design techniques. The author's clear, hands-on presentation and outstanding writing style help readers to better understand the material.· A Crash Course in Java· The Object-Oriented Design Process· Guidelines for Class Design· Interface Types and Polymorphism· Patterns and GUI Programming· Inheritance and Abstract Classes· The Java Object Model· Frameworks· Multithreading· More Design Patterns

Head First Software Development

Understand Gang of Four, architectural, functional, and reactive design patterns and how to implement them on modern Java platforms, such as Java 12 and beyond Key Features Learn OOP, functional, and reactive patterns for creating readable and maintainable code Explore architectural patterns and practices for building scalable and reliable applications Tackle all kinds of performance-related issues and streamline development using design patterns Book Description Java design patterns are reusable and proven solutions to software design problems. This book covers over 60 battle-tested design patterns used by developers to create functional, reusable, and flexible software. Hands-On Design Patterns with Java starts with an introduction to the Unified Modeling Language (UML), and delves into class and object diagrams with the help of detailed examples. You'll study concepts and approaches to object-oriented programming (OOP) and OOP design patterns to build robust applications. As you advance, you'll explore the categories of GOF design patterns, such as behavioral, creational, and structural, that help you improve code readability and enable large-scale reuse of software. You'll also discover how to work effectively with microservices and serverless architectures by using cloud design patterns, each of which is thoroughly explained and accompanied by real-world programming solutions. By the end of the book, you'll be able to speed up your software development process using the right design patterns, and you'll be comfortable working on scalable and maintainable projects of any size. What you will learn Understand the significance of design patterns for software engineering Visualize software design with UML diagrams Strengthen your understanding of OOP to create reusable software systems Discover GOF design patterns to develop scalable applications Examine programming challenges and the design patterns that solve them Explore architectural patterns for microservices and cloud development Who this book is for If you are a developer who wants to learn how to write clear, concise, and effective code for building production-ready applications, this book is for you. Familiarity with the fundamentals of Java is assumed.

Head First Object-Oriented Analysis and Design

This book, written by one of the designers of generics, is a thorough explanation of how to use generics, and particularly, the effect this facility has on the way developers use collections.

Clean Agile

Today, software engineers need to know not only how to program effectively but also how to develop proper engineering practices to make their codebase sustainable and healthy. This book emphasizes this difference between programming and software engineering. How can software engineers manage a living codebase that evolves and responds to changing requirements and demands over the length of its life? Based on their experience at Google, software engineers Titus Winters and Hyrum Wright, along with technical writer Tom Manshreck, present a candid and insightful look at how some of the world's leading practitioners construct and maintain software. This book covers Google's unique engineering culture, processes, and tools and how these aspects contribute to the effectiveness of an engineering organization. You'll explore three fundamental principles that software organizations should keep in mind when designing, architecting, writing, and maintaining code: How time affects the sustainability of software and how to make your code resilient over time How scale affects the viability of software practices within an engineering organization What trade-offs a typical engineer needs to make when evaluating design and development decisions

Design Patterns in Ruby (Adobe Reader)

You know you don't want to reinvent the wheel, so you look to design patterns—the lessons learned by those who've faced the same software design problems. With design patterns, you get to take advantage of the best practices and experience of others so that you can spend your time on something more challenging. Something more fun. This book shows you the patterns that matter, when to use them and why, how to apply them to your own designs, and the object-oriented design principles on which the patterns are based. Most importantly, you want to learn design patterns in a way that won't put you to sleep. If you've read a Head First book, you know what to expect—a visually rich format designed for the way your brain works. Newly updated and expanded, Head First Design Patterns, 2E, will load patterns into your brain in a way that sticks. In a way that makes you better at solving software design problems and better at speaking the language of patterns with others on your team. Join the hundreds of thousands of developers who have already improved their object-oriented design skills through Head First Design Patterns .

Head First Java

This book gets readers up to speed on the technology necessary to write servlets and JSPs, what makes the Container tick, how to use the new JSP Expression Language (EL), how to write deployment descriptors, and even how to use some server-side design patterns.

Understanding Machine Learning

Get the best out of Node.js by mastering its most powerful components and patterns to create modular and scalable applications with ease About This Book

Create reusable patterns and modules by leveraging the new features of Node.js . Understand the asynchronous single thread design of node and grasp all its features and patterns to take advantage of various functions. This unique guide will help you get the most out of Node.js and its ecosystem. Who This Book Is For The book is meant for developers and software architects with a basic working knowledge of JavaScript who are interested in acquiring a deeper understanding of how to design and develop enterprise-level Node.js applications. Basic knowledge of Node.js is also helpful to get the most out of this book. What You Will Learn Design and implement a series of server-side JavaScript patterns so you understand why and when to apply them in different use case scenarios Become comfortable with writing asynchronous code by leveraging constructs such as callbacks, promises, generators and the async-await syntax Identify the most important concerns and apply unique tricks to achieve higher scalability and modularity in your Node.js application Untangle your modules by organizing and connecting them coherently Reuse well-known techniques to solve common design and coding issues Explore the latest trends in Universal JavaScript, learn how to write code that runs on both Node.js and the browser and leverage React and its ecosystem to implement universal applications In Detail Node.js is a massively popular software platform that lets you use JavaScript to easily create scalable server-side applications. It allows you to create efficient code, enabling a more sustainable way of writing software made of only one language across the full stack, along with extreme levels of reusability, pragmatism, simplicity, and collaboration. Node.js is revolutionizing the web and the way people and companies create their software. In this book, we will take you on a journey across various ideas and components, and the challenges you would commonly encounter while designing and developing software using the Node.js platform. You will also discover the "Node.js way" of dealing with design and coding decisions. The book kicks off by exploring the basics of Node.js describing it's asynchronous single-threaded architecture and the main design patterns. It then shows you how to master the asynchronous control flow patterns, and the stream component and it culminates into a detailed list of Node.js implementations of the most common design patterns as well as some specific design patterns that are exclusive to the Node.js world. Lastly, it dives into more advanced concepts such as Universal Javascript, and scalability' and it's meant to conclude the journey by giving the reader all the necessary concepts to be able to build an enterprise grade application using Node.js. Style and approach This book takes its intended readers through a comprehensive explanation to create a scalable and efficient real-time server-side apps.

Effective DevOps

What will you learn from this book? This brain-friendly guide teaches you everything from JavaScript language fundamentals to advanced topics, including objects, functions, and the browser's document object model. You won't just be reading—you'll be playing games, solving puzzles, pondering mysteries, and interacting with JavaScript in ways you never imagined. And you'll write real code, lots of it, so you can start building your own web applications. Prepare to open your mind as you learn (and nail) key topics including: The inner details of JavaScript How JavaScript works with the browser The secrets of JavaScript types Using arrays The power of functions How to work with objects Making use of prototypes

Understanding closures Writing and testing applications What's so special about this book? We think your time is too valuable to waste struggling with new concepts. Using the latest research in cognitive science and learning theory to craft a multi-sensory learning experience, Head First JavaScript Programming uses a visually rich format designed for the way your brain works, not a text-heavy approach that puts you to sleep. This book replaces Head First JavaScript, which is now out of print.

Design Patterns in C#

Before GPS, before the compass, and even before cartography, humankind was navigating. A windswept tree, the depth of a puddle, or a trill of birdsong could point the way home-and, for the alert traveler, they still can. Whether you go exploring in the mountains or on a lunch break, natural navigation will keep you on course and open your eyes to the small wonders of the natural world. Almost anything in our environment can help us find our way-if we know what to look for. Adventurer and navigation expert Tristan Gooley unlocks the directional clues hidden in: the sun, moon, and stars clouds weather patterns lengthening shadows changing tides plant growth and the habits of local wildlife Enriched by helpful illustrations, and filled with navigational anecdotes collected across centuries, continents, and cultures, The Natural Navigator proves that anyone with a curious mind can still find south by looking at the moon-and find adventure in their own backyard.

[ROMANCE](#) [ACTION & ADVENTURE](#) [MYSTERY & THRILLER](#) [BIOGRAPHIES & HISTORY](#) [CHILDREN'S](#) [YOUNG ADULT](#) [FANTASY](#) [HISTORICAL FICTION](#) [HORROR](#) [LITERARY FICTION](#) [NON-FICTION](#) [SCIENCE FICTION](#)