Read Online Aspect Oriented Programming With Aspectj Ivan Kiselev Pdf File Free

OBJECT-ORIENTED PROGRAMMING WITH C++ *Object-oriented Programming in C++* **Beginning Object-Oriented Programming with C# Hands-On Object-Oriented Programming with Kotlin An Introduction to Object-Oriented Programming with Java** *Theoretical Aspects of Object-oriented Programming Advanced R* **Web Programming for Business OBJECT ORIENTED PROGRAMMING WITH C++ WITH EIGHTH EDITION** *Modern Programming: Object Oriented Programming and Best Practices* **Head First Object-Oriented Analysis and Design Learning Object-Oriented Programming** *Object-Oriented Programming with Turbo C++?* Object-Oriented Programming Languages and Event-Driven Programming Object-Oriented Programming with Visual Basic.NET **An Introduction to Object-Oriented Programming with Visual Basic .NET** C++ and Object-oriented Programming **Object Oriented Programming with Java** *Data Abstraction and Object-Oriented Programming in C++* **Understanding Object-oriented Programming with Java Object-oriented Programming with Prototypes** *Introduction to Object-Oriented Programming* A Comparative Presentation of Object Oriented Programming with C++ **Object-oriented Programming with Visual Basic .NET** Learning Object-Oriented Programming, Design and TDD with Pharo Object-Oriented Programming with QuickPascal The Interpretation of Object-Oriented Programming Languages **Java with Object-oriented Programming Aliasing in Object-Oriented Programming Object-oriented Programming in Python Object-oriented Programming Visual Object-oriented Programming Programming .NET Components** Python 3 Object Oriented Programming **Concepts of Object-Oriented Programming with Visual Basic** Object-oriented Programming with Smalltalk **Learn Object Oriented Programming Using Java: An UML based** Interactive Object Oriented Programming in Java Object Oriented Programming With C++ Beginning Java 17 Fundamentals

This book is the second edition of M.T. Somashekara's earlier book titled Programming in C++, under the new title Object-Oriented Programming with C++. In consonance with the new title, two chapters—one explaining the concepts of object-oriented programming and the other on object oriented software development—have been added, respectively, at the beginning and end of the book.

Substantial improvements have been effected in all chapters on C++. The book also carries a new chapter titled Standard Template Library. The book covers the C++ language thoroughly, from basic concepts through advanced topics such as encapsulation, polymorphism, inheritance, and exception handling. It presents C++ in a pedagogically sound way, giving many program examples to highlight the features and benefits of each of its concepts. The book is suitable for all engineering and science students including the students of computer applications for learning the C++ language from the first principles. KEY FEATURES : Logical flow of concepts starting from the preliminary topics to the major topics. Programs for each concept to illustrate its significance and scope. Complete explanation of each program with emphasis on its core segment. Chapter-end summary, review questions and programming exercises. Exhaustive glossary of programming terms. A Comparative Presentation of Object-Oriented Programming with C++ and Java, a comparative presentation of object-oriented programming with two of the most popular programming languages of today, teaches vital skills and techniques for the Internet age. Based on highly successful courses taught by the author, this book answers the need for a comprehensive educational program on the subject of object-oriented programming. In a clear and accessible format, the author compares and contrasts both languages, from basic language constructs to how both languages are used in application-level programming, such as graphics programming, network programming, and database programming. Learning to write a program in one language that corresponds to a given program in the other language enables students to tackle more difficult projects in either language. This book is intended as a serious introduction and reference for cutting-edge developers in the areas of visual and object-oriented programming. The first book on this topic, this guide focuses on the elements and strategies to help those who design visual object-oriented systems avoid some of the known pitfalls. While there are many books on particular languages, there are very few that deal with all aspects of object-oriented programming languages. The Interpretation of Object-Oriented Programming Languages provides a comprehensive treatment of the main approaches to object-oriented languages, including class-based, prototype and actor languages. This revised and extended edition includes a completely new chapter on Microsoft's new C# language, a language specifically designed for modern, component-oriented, networked applications. The chapter covers all aspects of C# that relate to object-oriented programming. It now also includes a new appendix on BeCecil, a kernel language that can implement object-oriented constructs within a single framework. An Essential Reference for Intermediate and Advanced R Programmers Advanced R presents useful tools and techniques for attacking many types of R programming problems, helping you avoid mistakes and dead ends. With more than ten years of experience programming in R, the author illustrates the elegance, beauty, and flexibility at the heart of R. The book develops the necessary skills to produce quality code that can be used in a variety of circumstances. You will learn: The fundamentals of R, including standard data types and functions Functional programming as a useful framework for solving wide classes of problems The positives and negatives of metaprogramming How to write fast, memory-efficient code This book not only helps current R users become R programmers but also shows existing programmers what’s special about R. Intermediate R

programmers can dive deeper into R and learn new strategies for solving diverse problems while programmers from other languages can learn the details of R and understand why R works the way it does. Provides information on analyzing, designing, and writing object-oriented software. As the title suggests, this book has two separate - though intertwined - goals: a description of the general concepts of object-orientation, and how to do object-oriented programming in Visual Basic. Readers are assumed to have no more than a familiarity with Visual Basic and some rudimentary knowledge of programming. Working on this premise, Steve Roman introduces the abstract concepts of object orientation, such as class, abstraction, and encapsulation, and then shows how each is implemented in a meaningful and useful application. He uses a hands-on style throughout: plenty of code is given and discussed, including error-handling. As a result, Visual Basic programmers and students will find this an invaluable introduction to the topic. Although the theory of object-oriented programming languages is far from complete, this book brings together the most important contributions to its development to date, focusing in particular on how advances in type systems and semantic models can contribute to new language designs.The fifteen chapters are divided into five parts: Objects and Subtypes, Type Inference, Coherence, Record Calculi, and Inheritance. The chapters are organized approximately in order of increasing complexity of the programming language constructs they consider - beginning with variations on Pascal- and Algol-like languages, developing the theory of illustrative record object models, and concluding with research directions for building a more comprehensive theory of object-oriented programming languages.Part I discusses the similarities and differences between "objects" and algebraic-style abstract data types, and the fundamental concept of a subtype. Parts II-IV are concerned with the "record model" of object-oriented languages. Specifically, these chapters discuss static and dynamic semantics of languages with simple object models that include a type or class hierarchy but do not explicitly provide what is often called dynamic binding. Part V considers extensions and modifications to record object models, moving closer to the full complexity of practical object-oriented languages.Carl A. Gunter is Professor in the Department of Computer and Information Science at the University of Pennsylvania. John C. Mitchell is Professor in the Department of Computer Science at Stanford University. An Introduction to Object-Oriented Programming with Java takes a full-immersion approach to object-oriented programming. Proper object-oriented design practices are emphasized throughout the book. Students learn how to use the standard classes first, then learn to design their own classes. Wu uses a gentler approach to teaching students how to design their own classes, separating the coverage into two chapters. GUI coverage is also located independently in the back of the book and can be covered if desired. Wu also features a robust set of instructors' materials including PowerPoint slides, code samples, and quiz questions. This book focuses on fundamental PHP coding, giving students practical, enduring skills to solve data and technical problems in business. Offering an innovative approach, it presents code that is clean, clearly explained and solutions-oriented. "An accessible introduction to the C++ language and object-oriented design for students and programmers who know at least one modern high-level language. Understanding that the greatest challenge in learning C++ is being able to think in terms of classes and objects, Kip Irvine introduces these topics immediately

as concepts in the context of real-world applications such as e-mail systems and automated bank tellers." "Through extensive use of short program examples and case studies, the author provides a concise, clear discussion of C++ syntax. He includes extensive coverage of the object model concept and how to use an object-oriented approach to design. Throughout the book, the importance of careful analysis and design of programs is evidenced."--BOOK JACKET.Title Summary field provided by Blackwell North America, Inc. All Rights Reserved "Object-Oriented Programming in C++" begins with the basic principles of the C++ programming language and systematically introduces increasingly advanced topics while illustrating the OOP methodology. While the structure of this book is similar to that of the previous edition, each chapter reflects the latest ANSI C++ standard and the examples have been thoroughly revised to reflect current practices and standards. Educational SupplementSuggested solutions to the programming projects found at the end of each chapter are made available to instructors at recognized educational institutions. This educational supplement can be found at www.prenhall.com, in the Instructor Resource Center. Dan Clark shows beginning VB.NET programmers how one goes about architecting an object oriented programming solution aimed at solving a business problem. Paul Wang's JAVA WITH OBJECT-ORIENTED PROGRAMMING eases students into an understanding of the object-oriented paradigm from the very first page, just as he does in JAVA WITH OBJECT-ORIENTED PROGRAMMING WITH WORLDWIDE WEB APPLICATIONS, on which this new book is modeled. After the early chapters that present classes and Java features and constructs, Wang introduces new object-oriented concepts throughout the book, while clearly showing how Java addresses these issues. He also goes the extra step of including case studies to illustrate how Java and object-oriented programming are applied. Early in the book, Wang introduces students to a case study involving a pocket calculator. This case study is revisited throughout the book as students learn new aspects of object-oriented programming and the Java language. The book then concludes with a chapter on some of the processes associated with object-oriented design. As a result, students are able to fully grasp the concepts they learn. Intended for the novice as well as for the experienced programmer who wants to learn more about object-oriented programming. Author is developer of the Omega programming environment. DLC: Object-oriented programming (Computer science) This latest addition to the family of Quick compilers contains several powerful and advanced features--object-oriented programming, a built-in debugger, and compatibility with Borland's Turbo Pascal. Illustrated. Learn Object Oriented Programming Using Java: An UML based Treatise with Live Examples from Science and Engineering Java is a language that is very well suited to object oriented programming and design. Java applications are stand-alone applications similar to the application we develop using other object oriented languages such as C++, .Net etc. This book describes the Java language as its purpose is to teach the basics of object-oriented programming. In this book, we introduce the java language as its design is clean and easy to follow. The creators of this language took a fresh, minimalistic perspective and approach towards the design of this language. They included only useful, indispensible features and eliminated features that do not add value or are redundant. This innovative approach taken by them makes Java an easier language to learn and master and provides an edge over

alternative programming languages. This language is an ideal, useful instrument to teach the fundamentals of object oriented programming and hence has been chosen for this book.This book covers the fundamentals of Java and object oriented programming and shows how well they work together. Several programming samples; tested against the latest Java version 8.0 have been provided. This book presents a balanced and flexible approach to the incorporation of object-oriented principles in introductory courses using Python. Familiarizes readers with the terminology of object-oriented programming, the concept of an object's underlying state information, and its menu of available behaviors. Includes an exclusive, easy-to-use custom graphics library that helps readers grasp both basic and more advanced concepts. Lays the groundwork for transition to other languages such as Java and C++. For those interested in learning more about object-oriented programming using Python. This book presents a survey of the state-of-the-art on techniques for dealing with aliasing in object-oriented programming. It marks the 20th anniversary of the paper The Geneva Convention On The Treatment of Object Aliasing by John Hogg, Doug Lea, Alan Wills, Dennis de Champeaux and Richard Holt. The 22 revised papers were carefully reviewed to ensure the highest quality.The contributions are organized in topical sections on the Geneva convention, ownership, concurrency, alias analysis, controlling effects, verification, programming languages, and visions. Compilers and applications programs are moving toward object-oriented programming (OOP), and C++ offers a more natural environment for OOP than any other language. Introduces programmers to OOP with Turbo C++. After describing OOP and the differences between Turbo C and Turbo C++ (OOP C), it puts Turbo C++ to work, applying object-oriented programming to numerous programming cases. Learn the fundamentals of the Java 17 LTS or Java Standard Edition version 17 Long Term Support release, including basic programming concepts and the object-oriented fundamentals necessary at all levels of Java development. Authors Kishori Sharan and Adam L. Davis walk you through writing your first Java program step-by-step. Armed with that practical experience, you'll be ready to learn the core of the Java language. Beginning Java 17 Fundamentals provides over 90 diagrams and 240 complete programs to help you learn the topics faster. While this book teaches you the basics, it also has been revised to include the latest from Java 17 including the following: value types (records), immutable objects with an efficient memory layout; local variable type inference (var); pattern matching, a mechanism for testing and deconstructing values; sealed types, a mechanism for declaring all possible subclasses of a class; multiline text values; and switch expressions. The book continues with a series of foundation topics, including using data types, working with operators, and writing statements in Java. These basics lead onto the heart of the Java language: object-oriented programming. By learning topics such as classes, objects, interfaces, and inheritance you'll have a good understanding of Java's object-oriented model. The final collection of topics takes what you've learned and turns you into a real Java programmer. You'll see how to take the power of object-oriented programming and write programs that can handle errors and exceptions, process strings and dates, format data, and work with arrays to manipulate data. What You Will Learn Write your first Java programs with emphasis on learning object-oriented programming How to work with switch expressions, value types (records),

local variable type inference, pattern matching switch and more from Java 17 Handle exceptions, assertions, strings and dates, and object formatting Learn about how to define and use modules Dive in depth into classes, interfaces, and inheritance in Java Use regular expressions Take advantage of the JShell REPL tool Who This Book Is For Those who are new to Java programming, who may have some or even no prior programming experience. 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. Discover object oriented programming with Java in this unique tutorial. This book uses Java and Eclipse to write and generate output for examples in topics such as classes, interfaces, overloading, and overriding. Interactive Object Oriented Programming in Java uniquely presents its material in a dialogue with the reader to encourage thinking and experimentation. Later chapters cover further Java programming concepts, such as abstract classes, packages, and exception handling.

At each stage you'll be challenged by the author to help you absorb the information and become a proficient Java programmer. Additionally, each chapter contains simple assignments to encourage you and boost your confidence level. What You Will Learn Become proficient in object oriented programming Test your skills in the basics of Java Develop as a Java programmer Use the Eclipse IDE to write your code Who This Book Is For Software developers and software testers. Essential concepts of programming language design and implementation are explained and illustrated in the context of the object-oriented programming language (OOPL) paradigm. Written with the upper-level undergraduate student in mind, the text begins with an introductory chapter that summarizes the essential features of an OOPL, then widens the discussion to categorize the other major paradigms, introduce the important issues, and define the essential terms. After a brief second chapter on event-driven programming (EDP), subsequent chapters are built around case studies in each of the languages Smalltalk, C++, Java, C#, and Python. Included in each case study is a discussion of the accompanying libraries, including the essential container classes. For each language, one important event-driven library is singled out and studied. Sufficient information is given so that students can complete an event-driven project in any of the given languages. After completing the course the student should have a solid set of skills in each language the instructor chooses to cover, a comprehensive overview of how these languages relate to each other, and an appreciation of the major issues in OOPL design. Key Features: •Provides essential coverage of Smalltalk origins, syntax, and semantics, a valuable asset for students wanting to understand the hybrid Objective C language •Provides detailed case studies of Smalltalk, Java, C++, C#, and Python and features a side-by-side development of the Java and C++ languages--highlighting their similarities and differences •Sets the discussion in a historical framework, tracing the roots of the OOPLs back to Simula 67. •Provides broad-based coverage of all languages, imparting essential skills as well as an appreciation for each language's design philosophy •Includes chapter summary, review questions, chapter exercises, an appendix with event-driven projects, and instructor resources We are living in the world that is moving from the asset based economy to knowledge based economy. Our thinking process is changing from local scope to global scope. Programming is not an exception for paradigm shift. It is changing from modules to objects. And now it is your turn for shifting from C to C++. C++ is a super set of C language. It provides the C programmers the flavor of OOPS. With its object-oriented programming features like encapsulation, inheritance and polymorphism, C++ offers a number of benefits over C language. Object-Oriented Programming with C++ is a book also designed as per the syllabus of IV semester B.E. (Computer Science & Engineering and Information Science Engineering) course framed by the Visveswaraiah Technological University, Belgaum. This book is to teach the students the object-oriented programming concepts and C++. This book is written in a easy, riveting and readable style. The information provided in the book is helpful for B.E., B.Sc., BCA, MCA and M.Tech students of all universities The book provides around 200 programs to enrich the better understanding of C++. All C++ programming lab assignments are provided in Appendix-A. All the programs have been run and tested on Turbo C++ compiler on MS-DOS. However, some programs hardly countable with fingers are executed on Borland's C++ compiler. These programs are

exclusively mentioned with the comment -This program is run on Borland’s C++. 'Programming .NET Components', second edition, updated to cover .NET 2.0., introduces the Microsoft .NET Framework for building components on Windows platforms. From its many lessons, tips, and guidelines, readers will learn how to use the .NET Framework to program reusable, maintainable, and robust components. The ideal beginner's guide to C# and object-oriented programming Wrox beginners' guides have the perfect formula for getting programming newcomers up and running. This one introduces beginners to object-oriented programming using C# to demonstrate all of the core constructs of this programming framework. Using real-world situations, you'll discover how to create, test, and deliver your programs and how to work with classes, arrays, collections, and all the elements of object-oriented programming. Covers exactly what beginners, even those with no prior programming experience, need to know to understand object-oriented programming and start writing programs in C# Explains the advantages and disadvantages of C#, and tips for understanding C# syntax Explores properties, encapsulation, and classes; value data types; operands and operators; errors and debugging; variables; and reference types Shows how to use statement repetition and program loops, understand arrays and collections, and write your own classes Also covers inheritance and polymorphism Beginning Object-Oriented Programming with C# uses the tried-and-true Wrox formula for making this popular programming method easy to learn. Filmed work by students of the School of Design, Swinburne University of Technology. Harness the power of Python 3 objects. Learning Object-Oriented Programming is an easy-to-follow guide full of hands-on examples of solutions to common problems with object-oriented code in Python, JavaScript, and C#. It starts by helping you to recognize objects from real-life scenarios and demonstrates that working with them makes it simpler to write code that is easy to understand and reuse. You will learn to protect and hide data with the data encapsulation features of Python, JavaScript, and C#. You will explore how to maximize code reuse by writing code capable of working with objects of different types, and discover the advantage of duck typing in both Python and JavaScript, while you work with interfaces and generics in C#. With a fair understanding of interfaces, multiple inheritance, and composition, you will move on to refactor existing code and to organize your source for easy maintenance and extension. Learning Object-Oriented Programming will help you to make better, stronger, and reusable code. Learn everything you need to know about object-oriented programming with the latest features of Kotlin 1.3 Key FeaturesA practical guide to understand objects and classes in KotlinLearn to write asynchronous, non-blocking codes with Kotlin coroutinesExplore Encapsulation, Inheritance, Polymorphism, and Abstraction in KotlinBook Description Kotlin is an object-oriented programming language. The book is based on the latest version of Kotlin. The book provides you with a thorough understanding of programming concepts, object-oriented programming techniques, and design patterns. It includes numerous examples, explanation of concepts and keynotes. Where possible, examples and programming exercises are included. The main purpose of the book is to provide a comprehensive coverage of Kotlin features such as classes, data classes, and inheritance. It also provides a good understanding of design pattern and how Kotlin syntax works with object-oriented techniques. You will also gain familiarity with syntax in this book by

writing labeled for loop and when as an expression. An introduction to the advanced concepts such as sealed classes and package level functions and coroutines is provided and we will also learn how these concepts can make the software development easy. Supported libraries for serialization, regular expression and testing are also covered in this book. By the end of the book, you would have learnt building robust and maintainable software with object oriented design patterns in Kotlin. What you will learnGet an overview of the Kotlin programming languageDiscover Object-oriented programming techniques in Kotlin Understand Object-oriented design patternsUncover multithreading by Kotlin wayUnderstand about arrays and collectionsUnderstand the importance of object-oriented design patternsUnderstand about exception handling and testing in OOP with KotlinWho this book is for This book is for programmers and developers who wish to learn Object-oriented programming principles and apply them to build robust and scalable applications. Basic knowledge in Kotlin programming is assumed Michael McMillan provides a complete presentation of the object-oriented features of the Visual Basic .NET language for advanced Visual Basic programmers. Beginning with an introduction to abstract data types and their initial implementation using structures, he explains standard OOP topics including class design, inheritance, access modifiers and scoping issues, abstract classes, design and implemention of interfaces and design patterns, and refactoring in VB.NET. More advanced OOP topics are included as well, such as reflection, object persistence, and serialization. To tie everything together, McMillan demonstrates sound OOP design and implementation principles through practical examples of standard Windows applications, database applications using ADO.NET, Web-based applications using ASP.NET, and Windows service applications. Software -- Programming Languages. A programmer's complete guide to Visual Basic .NET. Starting with a sample application and a high-level map, the book jumps right into showing how the parts of .NET fit with Visual Basic .NET. Topics include the common language runtime, Windows Forms, ASP.NET, Web Forms, Web Services, and ADO.NET. Object oriented programming is a way of thinking about problems. Smalltalk is one of the purest incarnations of an object-oriented programming language. Using a pedagogical approach, this book covers all aspects of object oriented programming: first through the study of various preexisting Smalltalk classes, their implementation and use; then through a detailed description of an implementation of an interactive Lindenmayer system and through implementation of a series of calculators. The author addresses such subjects as graphics programming, dependency mechanisms and hierarchical specialization. This book fills the gap for an in-depth self-study reference, permitting the reader to master all aspects of object-oriented programming through a large set of exercises with highly detailed resources. Downloadable software content for practice applications Covers all aspects of Smalltalk: theconcepts of primitive objects, classes and instances, static and dynamic inheritance and methods, as well as graphical programming, the dependency mechanisms and the handling of exceptions Features in-depth studies of two programming projects and annotated solutions to all exercies and appendices

- OBJECT ORIENTED PROGRAMMING WITH C
- Object oriented Programming In C
- Beginning Object Oriented Programming With C
- Hands On Object Oriented Programming With Kotlin
- An Introduction To Object Oriented Programming With Java
- Theoretical Aspects Of Object oriented Programming
- Advanced R
- Web Programming For Business
- OBJECT ORIENTED PROGRAMMING WITH C WITH EIGHTH EDITION
- Modern Programming Object Oriented Programming And Best Practices
- Head First Object Oriented Analysis And Design
- Learning Object Oriented Programming
- Object Oriented Programming With Turbo C
- Object Oriented Programming Languages And Event Driven Programming
- Object Oriented Programming With Visual BasicNET
- An Introduction To Object Oriented Programming With Visual Basic NET
- C And Object oriented Programming
- Object Oriented Programming With Java
- Data Abstraction And Object Oriented Programming In C
- Understanding Object oriented Programming With Java
- Object oriented Programming With Prototypes
- Introduction To Object Oriented Programming
- A Comparative Presentation Of Object Oriented Programming With C
- Object oriented Programming With Visual Basic NET
- Learning Object Oriented Programming Design And TDD With Pharo
- Object Oriented Programming With QuickPascal
- The Interpretation Of Object Oriented Programming Languages
- Java With Object oriented Programming

- Aliasing In Object Oriented Programming
- Object oriented Programming In Python
- Object oriented Programming
- Visual Object oriented Programming
- Programming NET Components
- Python 3 Object Oriented Programming
- Concepts Of Object Oriented Programming With Visual Basic
- Object oriented Programming With Smalltalk
- Learn Object Oriented Programming Using Java An UML Based
- Interactive Object Oriented Programming In Java
- Object Oriented Programming With C
- Beginning Java 17 Fundamentals