

Programming Smalltalk Object Orientation From The Beginning An Introduction To The Principles Of Programming

An introduction to programming in Smalltalk, covering technical background for programmers and managers and introducing some of the basic philosophy of the language. Step-by-step instructions take the reader through the basics via object-oriented programming with the Smalltalk language and its development environment. Includes a tour of the Smalltalk class library and the model-view-controller mechanism. For programmers who want to move from traditional languages to an object-oriented language. Annotation copyright by Book News, Inc., Portland, OR. The continual evolution of object oriented technologies creates both opportunities and challenges. However, despite the growing popularity of object oriented technology, there are numerous issues that have contributed to its inability to firmly entrench itself and take over for the older, proven technologies. Object oriented technology's image problem has created a highly difficult decision making process for corporations considering widespread adoption of these technologies. Object Oriented Technologies: Opportunities and Challenges addresses concerns, opportunities and technology trends in the application of object oriented technologies. The chapters of this book were selected to represent a variety of perspectives concerning the present and future of this broad sub-field of software development. Object-oriented programming originated with the Simula language developed by Kristen Nygaard in Oslo in the 1960s. Now, from the birthplace of OOP, comes the new BETA programming language, for which this book is both tutorial and reference. It provides a clear introduction to the basic concepts of OOP and to more advanced topics.

There has been an increasing demand in GIS for systems that support historical data: time-series data as well as mobility information. From a modelling perspective, there are advantages in integrating object-oriented analysis and design to databases as well as to visualisation capabilities of GIS. Object-Oriented Design for Temporal GIS explores the major components of the object-oriented analysis and design methods, how they can be used for modelling spatio-temporal data, and how these components are developed and maintained within a GIS. It also offers practical guidance to object-oriented methods by demonstrating the feasibility of applying such methods to issues involved in handling spatio-temporal data. The author demonstrates how this knowledge might be used in a wide range of applications such as political boundary record maintenance (historical data), disease incidence rate analysis in epidemics (diffusion rate), and environmental studies of climate change (time-series data). This understanding contributes to the development of theory in GIS and improves the design of GIS to support the modelling of semantics, space and time elements of geographical information.

Object-oriented Programming in the BETA Programming Language

Programming Smalltalk - Object-Oriented from the Beginning

An Evolutionary Approach

ECOOP 2004 - Object-Oriented Programming

Object Orientation

Java for Practitioners

For more than 40 years, Computerworld has been the leading source of technology news and information for IT influencers worldwide. Computerworld's award-winning Web site (Computerworld.com), twice-monthly publication, focused conference series and custom research form the hub of the world's largest global IT media network.

Walks COBOL users through the next phase of COBOL: Object-Oriented COBOL. Teaches how to integrate COBOL with object-oriented methodologies.

Software quality is a generalised statement difficult to agree or disagree with until a precise definition of the concept of "Software Quality" is reached in terms of measurable quantities. Unfortunately, for the software technology the basic question of: • what to measure; • how to measure; • when to measure; • how to deal with the data obtained are still unanswered and are also closely dependent on the field of application. In the past twenty years or more there have been a number of conferences and debates focusing on the concept of Software Quality, which produced no real industrial impact. Recently, however, the implementation of a few generic standards (ISO 9000, IEEE etc.) has produced and improved application of good practice principles at the industrial level. As a graduate in PHYSICS, I still believe it is a long way before the concept of Software Quality can be defined exactly and measured, if ever. This is why I think the AQUiS series of conferences is important, its object begin to provide a platform for the transfer of technology and know how between Academic, Industrial and Research Institutions, in the field of Software Quality. Their objects are: • to provide a forum for the introduction and discussion of new research breakthroughs in Software Quality; • to provide professional Software Quality engineers with the necessary exposure to the results of current research; • to expose the research community to the problems of practical application of new results.

Fritzens covers the Modella language in impressive depth from the basic concepts such as cyber-physical, equation-base, object-oriented, system, model, and simulation, while also incorporating over a hundred exercises and their solutions for a tutorial, easy-to-read experience. The only book with complete Modella 3.3 coverage Over one hundred exercises and solutions Examines basic concepts such as cyber-physical, equation-based, object-oriented, system, model, and simulation

Concurrent Object-Oriented Programming and Petri Nets

Reliable Object-Oriented Software

15th European Conference, Glasgow, UK, July 25-29, 2005. Proceedings

From GKS and PHIGS to Object-Oriented Systems

Object-Oriented COBOL

Smalltalk and Object Orientation

Introduction: What does it mean to be object-oriented, anyway? Object-orientation - Who ordered that? Object-oriented design notation. The basic notation for classes em methods. Inheritance and aggregation diagrams. The object-communication diagram. State-transition diagrams. Additional OODN diagrams. The principles of object-oriented design: Encapsulation and connascence. Domains, encumbrance, and cohesion.

Properties of classes and subclasses. The perils of inheritance and polymorphism. Class interfaces. Appendix A: Checklist for an object-oriented design walkthrough. Appendix B: The Object-oriented design owner's manual. Appendix C: Blitz guide to object-oriented terminology.

If you're considering R for statistical computing and data visualization, this book provides a quick and practical guide to just about everything you can do with the open source R language and software environment. You'll learn how to write R functions and use R packages to help you prepare, visualize, and analyze data. Author Joseph Adler illustrates each process with a wealth of examples from medicine, business, and sports.

Updated for R 2.14 and 2.15, this second edition includes new and expanded chapters on R performance, the ggplot2 data visualization package, and parallel R computing with Hadoop. Get started quickly with an R tutorial and hundreds of examples Explore R syntax, objects, and other language details Find thousands of user-contributed R packages online, including Bioconductor Learn how to use R to prepare data for analysis

Visualize your data with R's graphics, lattice, and ggplot2 packages Use R to calculate statistical tests, fit models, and compute probability distributions Speed up intensive computations by writing parallel R programs for Hadoop Get a complete desktop reference to R

Programming in an Object-Oriented Environment provides an in-depth look at the concepts behind the technology of object-oriented programming. This book explains why object-oriented programming has the potential to vastly improve the productivity of programmers and how to apply this technology in a practical environment. Many programming examples are included, focusing on how different programming languages support the core of object-oriented concepts. C++ is used as the main sample language throughout this text. This monograph consists of two major parts. Part I provides an introduction to object-oriented concepts, their rationale and their implementation in programming languages. The object-oriented approach to programming in an object-oriented environment is discussed in Part II. This publication is intended for software professionals who are interested in learning the fundamental concepts of object-oriented programming and how to apply these concepts in a practical computer environment.

"..... object oriented seems to be becoming in the 1980s what structured programming was in the 1970s." - Brian Randell and Pete Lee This quotation is from the invitation to the annual Newcastle University Conference on Main Trends in Computing, September 1988. It seems to capture the situation quite well, only that the object orientation is being materialised in languages and language constructs, as well as in the style of programming and as a perspective upon the task considered. The second European Conference on Object Oriented Programming (ECOOP '88) was held in Oslo, Norway, August 15-17, 1988, in the city where object oriented programming was born more than 20 years ago, when the Simula language appeared. The objectives of ECOOP '88 were to present the best international work in the field of object oriented programming to interested participants from industry and academia, and to be a forum for the exchange of ideas and the growth of professional relationships.

Object-Oriented Graphics

Principles of Object-Oriented Modeling and Simulation with Modella 3.3

Advances in Petri Nets

ECOOP 2005 - Object-Oriented Programming

Programming in an Object-Oriented Environment

Object Oriented Technologies: Opportunities and Challenges

Principles of the Book This book presents an approach to improve the standard object-oriented programming model. The proposal is aimed at supporting a larger range of incremental behavior variations and thus promises to be more effective in mastering the complexity of today's software. The ability of dealing with the evolutionary nature of software is one of main merits of object-oriented data abstraction and inheritance. Object-orientation allows to organize software in a structured way by separating the description of different kinds of an abstract data type into different classes and loosely connecting them by the inheritance hierarchy. Due to this separation, the software becomes free of conditional logics previously needed for distinguishing between different kinds of abstractions and can thus more easily be incrementally extended to support new kinds of abstractions. In other words, classes and inheritance are means to properly model variations of behavior related to the existence of different kinds of an abstract data type. The support for extensibility and reuse with respect to such kind-specific behavior variations is among the main reasons for the increasing popularity of object-oriented programming in the last two decades. However, this popularity does not prevent us from questioning the real effectiveness of current object-oriented techniques in supporting incremental variations. In fact, this popularity makes a critical investigation of the variations that can actually be performed incrementally even more important.

"There are few books that show how to build programs of any kind. One common theme is compiler building, and there are shelves full of them. There are few others. It's an area, or a void, that needs filling. This book does a great job of showing how to build numerical analysis programs." -David N. Smith, IBM T J Watson Research Center Numerical methods naturally lend themselves to an object-oriented approach. Mathematics builds high-level ideas on top of previously described, simpler ones. Once a property is demonstrated for a given concept, it can be applied to any new concept sharing the same premise as the original one, similar to the ideas of reuse and inheritance in object-oriented (OO) methodology. Few books on numerical methods teach developers much about designing and building good code. Good computing routines are problem-specific. Insight and understanding are what is needed, rather than just recipes and black box routines. Developers need the ability to construct new programs for different applications. Object-Oriented Implementation of Numerical Methods reveals a complete OO design methodology in a clear and systematic way. Each method is presented in a consistent format, beginning with a short explanation and following with a description of the general OO architecture for the algorithm. Next, the code implementations are discussed and presented along with real-world examples that the author, an experienced numerical analyst, has used in a variety of commercial applications. Features: Reveals the design methodology behind the code, including design patterns where appropriate, rather than just presenting canned solutions. Implements all methods side by side in both Java and Smalltalk. This contrast can significantly enhance your understanding of the nature of OO programming languages. Provides a step-by-step pathway to new object-oriented techniques for programmers familiar with using procedural languages such as C or Fortran for numerical methods. Includes a chapter on data mining, a key application of numerical methods.

Computer systems play an important role in our society. Software drives these systems. Massive investments of time and resources are made in developing and implementing these systems. Maintenance is inevitable. It is hard and costly. Considerable resources are required to keep the systems active and dependable. We cannot maintain software unless maintainability characters are built into the products and processes. There is an urgent need to reinforce software development practices based on quality and reliability principles. Though maintenance is a mini development lifecycle, it has its own problems. Maintenance issues need corresponding tools and techniques to address them. Software professionals are key players in maintenance. While development is an art and science, maintenance is a craft. We need to develop maintenance personnel to master this craft. Technology impact is very high in systems world today. We can no longer conduct business in the way we did before. That calls for reengineering systems and software. Even reengineered software needs maintenance, soon after its implementation. We have to take business knowledge, procedures, and data into the newly reengineered world. Software maintenance people can play an important role in this migration process. Software technology is moving into global and distributed networking environments. Client/server systems and object-orientation are on their way. Massively parallel processing systems and networking resources are changing database services into corporate data warehouses. Software engineering environments, rapid application development tools are changing the way we used to develop and maintain software. Software maintenance is moving from code maintenance to design maintenance, even onto specification maintenance. Modifications today are made at specification level, regenerating the software components, testing and integrating them with the system. Eventually software maintenance has to manage the evolution and evolutionary characteristics of software systems. Software professionals have to maintain not only the software, but the momentum of change in systems and software. In this study, we observe various issues, tools and techniques, and the emerging trends in software technology with particular reference to maintenance. We are not searching for specific solutions. We are identifying issues and finding ways to manage them, live with them, and control their negative impact.

This book was originally written to support an introductory course in Object Orientation through the medium of Smalltalk (and VisualWorks in particular). However, it can be used as a book to teach the reader Smalltalk, to introduce object orientation as well as present object oriented design and analysis. It takes as its basic premise that most Computer Scientists I Software Engineers learn best by doing rather than from theoretical notes. The chapters therefore attempt to introduce concepts by getting you the reader to do things, rather than by extensive theoretical discussions. This means that these chapters take a hands-on approach to the subject and assume that the student/reader has a suitable Small talk environment available to them. The chapters are listed below and are divided into six parts. The reader is advised to work through Parts 1 and 3 thoroughly in order to gain a detailed understanding of object orientation. Part 2 then provides an introduction to the Smalltalk environment and language. Other chapters may then be dipped into as required. For example, if the reader wishes to hone their Smalltalk skills then the chapters in Part 4 would be useful. However, if at that point they wish to get on and discover the delights of graphical user interfaces in Smalltalk, then Part 5 could be read next. Part 6 presents some more advanced subjects such as metaclasses and concurrency which are not required for straight forward Small talk development.

R in a Nutshell

ECOOP '87. European Conference on Object-Oriented Programming

ECOOP 2009 -- Object-Oriented Programming

What Every Programmer Should Know about Object-oriented Design

Object-Oriented Programming

Object-Oriented and Mixed Programming Paradigms

This book shows readers how to use the most of CF using Object Orientation. The author takes a hands-on approach to learning CF and object orientation, using lots of worked examples. The text provides an ideal base from which to start programming. After introducing the CF language and object orientation, John Hunt goes on to explain: how to construct a user interface for a simple editor; how to obtain information on files and directories and how objects can be stored and restored using serialization...Presents CF and object-orientation as a coherent whole, using one to strengthen the presentation of the other -Includes lots of complete and worked examples to clarify readers'understanding -The source code for the examples is available at: <http://www.guide-to-csharp.net> -Hunt is a successful Springer author, and this book is written in the same style as his Java for Practitioners

More than a guide to the Smalltalk language.

At present, object-oriented programming is emerging from the research laboratories and invading into the field of industrial applications. More and more products have been implemented with the aid of object-oriented programming techniques and tools, usually as extensions of traditional languages in hybrid development systems. Some of the better known examples are OSF-Motif, News, Objective-C on the NeXT computer, the C extension C++, and CLOS an object oriented extension of LISP. All of these developments incorporate interactive graphics. Effective object-oriented systems in combination with a graphics kernel does it mean that the field of computer graphics has now become merely an aspect of the object-oriented world? We do not think so. In spite of interesting individual developments, there are still no sound object-oriented graphics systems available. If it is desired to develop a complex graphics application embed ded in a window-oriented system then it is still necessary to work with element tury tools. What is to be deployed and interactively modified inside a window must be specified with a set of graphics primitives at a low level, or has to be written with a standardized graphics kernel system such as GKS or PHIGS, i. e., by kernels specified and implemented in a non-object-oriented style. With the terms GKS and PHIGS we enter the world of international graphics standards. GKS and PHIGS constitute systems, not mere collections of graphics primitives.

* Treats LISP as a language for commercial applications, not a language for academic AI concerns. This could be considered to be a secondary text for the Lisp course that most schools teach. This would appeal to students who sat through a LISP course in college without quite getting it - so a "nostalgia" approach, as in "now-lisp can be practical...". * Discusses the Lisp programming model and environment. Contains an introduction to the language and gives a thorough overview of all of Common Lisp's main features. * Designed for experienced-programmers no matter what languages they may be coming from and written for a modern audience--programmers who are familiar with languages like Java, Python, and Perl. * Includes several examples of working code that actually does something useful like Web programming and database access.

Smalltalk, Objects, and Design

ECOOP '88 European Conference on Object-Oriented Programming

EBOOK: OBJECT-ORIENTED SOFTWARE

A Cyber-Physical Approach

The Interpretation of Object-Oriented Programming Languages

New Directions in Computer Graphics

ECOOP is the premier forum in Europe for bringing together practitioners, -sachers, and students to share their ideas and experiences in a broad range of disciplines woven with the common thread of object technology. It is a collage of events, including outstanding invited speakers, carefully refereed technical -pers, practitioner reports reflecting real-world experience, panels, topic-focused workshops, demonstrations, and an interactive posters session. The 18th ECOOP 2004 conference held during June 14-18, 2004 in Oslo, Norway represented another year of continued success in object-oriented programming, both as a topic of academic study and as a vehicle for industrial software development. Object-oriented technology has come of age; it is now the commonly established method for most software projects. However, an -panding field of applications and new technological challenges provide a strong demand for research in object-oriented design and programming methods, as well as implementation techniques. There is also an increasing interest in the integration of object-orientation with other software development techniques. We anticipate therefore that object-oriented programming will be a fruitful subject of research for many years to come. This year, the program committee received 322 submissions, of which 25 were accepted for publication after thorough reviewing process. Every paper received at least 4 reviews. Papers were evaluated based on relevance, significance, clarity, originality, and correctness. The topics covered include: programming concepts, program analysis, software engineering, aspects and components, middleware, verification, systems and implementation techniques. These were complemented by two invited talks, from Matthias Felleisen and Tom Henzinger. Their titles and abstracts are also included in these proceedings.

This book provides a comprehensive treatment of the main approaches to object-oriented programming, including class-based programming, prototype programming, and actor-like languages. This book will be useful for students studying object-oriented programming, as well as for researchers and computer scientists requiring a detailed account of object-oriented programming languages and their central concepts. Programming Smalltalk - Object-Orientation from the BeginningAn introduction to the principles of programmingSpringer

Filed work by students of the School of Design, Swinburne University of Technology.

Opportunities and Challenges

An Introduction with Java & Smalltalk

Concepts, Analysis & Design, Languages, Databases, Graphical User Interfaces, Standards

Object-Oriented Implementation of Numerical Methods

(Issues, Tools, Techniques, and Trends)

Achieving Quality in Software

Object orientation has become a "must know" subject for managers, researchers, and software practitioners interested in the design, evolution, reuse and management of efficient software components. The book contains technical papers reflecting both theoretical and practical contributions from researchers in the field of object-oriented (OO) databases and software engineering systems. The book identifies actual and potential areas of integration of OO and database technologies, current and future research directions in software methodologies, and reflections about the OO paradigm. In providing current research and relevant information about this promising and rapidly growing field of object-oriented databases and software engineering systems, this book is invaluable to research scientists, practitioners, and graduate students working in the areas of databases and software engineering.

A straightforward, step-by-step introduction to clear and elegant object-oriented programming. Using a language that's perfect for this kind of programming, the book has been tested in numerous courses and workshops over ten years. Programming Smalltalk is particularly suited for readers with no prior programming knowledge. Starting from the first principles of programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies.

Quickly acquire the knowledge and skills you need to use object technology in your next development project A practical, down-to-earth introduction to object-oriented terms, concepts, and technologies, Object Orientation, Second Edition is for developers and programmers who are eager to start using object-oriented technology right away. Building step-by-step from the fundamentals to advanced design and development topics, this book supplies you with all the in-depth technical information and guidance you need to confidently incorporate object-oriented tools and techniques into your next project, no matter what your level of experience. Thanks to the authors' clear, straightforward explanations and professional insights, as well as the many real-world examples appearing throughout the book, you'll quickly acquire a solid working knowledge of • Abstract data typing, inheritance, and identity • Object-oriented analysis and design-including Booch, Rumbaugh, and other OOA and OOD methodologies • Object-oriented programming languages-including the object-oriented features of C++, Smalltalk, Ada, Eiffel, and other languages • Object-oriented database management systems-including OOODB, ORDB, client/server concepts, and examples from Object Design, Gem Stone, Versant, UniSQL, Objectivity, ODB-II, and other systems • Object-oriented GUI design-including explanations of Visual C++ and Foundation Classes, MacAPP, and NeXTStep • Object sharing and interchange with OLE 2 and OpenDoc • OMA, ODMG-93, and other object-oriented standardization efforts • And much more Featuring over 50% new and revised material, this Second Edition of Setrag Khoshafian and Razmik Abnous's bestselling is now more than ever the best practical introduction to object technology for programmers and developers.

This volume contains the proceedings of the first European Conference on Object-Oriented Programming, held in Paris, June 15-17, 1987. The idea of this annual conference series is to provide a forum for theorists and practitioners interested in the object-oriented programming paradigm. The contributions cover the following aspects of object-oriented programming: methodology, implementation, Petri, interfaces, languages, simulation, inheritance.

Object-oriented Technology For Database And Software Systems

Software Maintenance - A Management Perspective

Journal of Object-oriented Programming

18th European Conference, Oslo, Norway, June 14-18, 2004, Proceedings

Library of Congress Subject Headings

A straightforward, step-by-step introduction to clear and elegant object-oriented programming. Using a language that's perfect for this kind of programming, the book has been tested in numerous courses and workshops over ten years. Programming Smalltalk is particularly suited for readers with no prior programming knowledge. Starting from the first principles of programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies. Not only is this book a step-by-step introduction to clear and elegant object-oriented programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies.

Quickly acquire the knowledge and skills you need to use object technology in your next development project A practical, down-to-earth introduction to object-oriented terms, concepts, and technologies, Object Orientation, Second Edition is for developers and programmers who are eager to start using object-oriented technology right away. Building step-by-step from the fundamentals to advanced design and development topics, this book supplies you with all the in-depth technical information and guidance you need to confidently incorporate object-oriented tools and techniques into your next project, no matter what your level of experience. Thanks to the authors' clear, straightforward explanations and professional insights, as well as the many real-world examples appearing throughout the book, you'll quickly acquire a solid working knowledge of • Abstract data typing, inheritance, and identity • Object-oriented analysis and design-including Booch, Rumbaugh, and other OOA and OOD methodologies • Object-oriented programming languages-including the object-oriented features of C++, Smalltalk, Ada, Eiffel, and other languages • Object-oriented database management systems-including OOODB, ORDB, client/server concepts, and examples from Object Design, Gem Stone, Versant, UniSQL, Objectivity, ODB-II, and other systems • Object-oriented GUI design-including explanations of Visual C++ and Foundation Classes, MacAPP, and NeXTStep • Object sharing and interchange with OLE 2 and OpenDoc • OMA, ODMG-93, and other object-oriented standardization efforts • And much more Featuring over 50% new and revised material, this Second Edition of Setrag Khoshafian and Razmik Abnous's bestselling is now more than ever the best practical introduction to object technology for programmers and developers.

This volume contains the proceedings of the first European Conference on Object-Oriented Programming, held in Paris, June 15-17, 1987. The idea of this annual conference series is to provide a forum for theorists and practitioners interested in the object-oriented programming paradigm. The contributions cover the following aspects of object-oriented programming: methodology, implementation, Petri, interfaces, languages, simulation, inheritance.

Object-Oriented and Mixed Programming Paradigms

Object-Oriented Implementation of Numerical Methods

(Issues, Tools, Techniques, and Trends)

Achieving Quality in Software

Object orientation has become a "must know" subject for managers, researchers, and software practitioners interested in the design, evolution, reuse and management of efficient software components. The book contains technical papers reflecting both theoretical and practical contributions from researchers in the field of object-oriented (OO) databases and software engineering systems. The book identifies actual and potential areas of integration of OO and database technologies, current and future research directions in software methodologies, and reflections about the OO paradigm. In providing current research and relevant information about this promising and rapidly growing field of object-oriented databases and software engineering systems, this book is invaluable to research scientists, practitioners, and graduate students working in the areas of databases and software engineering.

A straightforward, step-by-step introduction to clear and elegant object-oriented programming. Using a language that's perfect for this kind of programming, the book has been tested in numerous courses and workshops over ten years. Programming Smalltalk is particularly suited for readers with no prior programming knowledge. Starting from the first principles of programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies. Not only is this book a step-by-step introduction to clear and elegant object-oriented programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies. Now translated into English, this edition was completely revised to be consistent with the latest version of Cincom® VisualWorks®, a professional Smalltalk environment. All examples were created using VisualWorks, which is available without cost for educational purposes, and can be downloaded and installed on any up-to-date computer.

Quickly acquire the knowledge and skills you need to use object technology in your next development project A practical, down-to-earth introduction to object-oriented terms, concepts, and technologies, Object Orientation, Second Edition is for developers and programmers who are eager to start using object-oriented technology right away. Building step-by-step from the fundamentals to advanced design and development topics, this book supplies you with all the in-depth technical information and guidance you need to confidently incorporate object-oriented tools and techniques into your next project, no matter what your level of experience. Thanks to the authors' clear, straightforward explanations and professional insights, as well as the many real-world examples appearing throughout the book, you'll quickly acquire a solid working knowledge of • Abstract data typing, inheritance, and identity • Object-oriented analysis and design-including Booch, Rumbaugh, and other OOA and OOD methodologies • Object-oriented programming languages-including the object-oriented features of C++, Smalltalk, Ada, Eiffel, and other languages • Object-oriented database management systems-including OOODB, ORDB, client/server concepts, and examples from Object Design, Gem Stone, Versant, UniSQL, Objectivity, ODB-II, and other systems • Object-oriented GUI design-including explanations of Visual C++ and Foundation Classes, MacAPP, and NeXTStep • Object sharing and interchange with OLE 2 and OpenDoc • OMA, ODMG-93, and other object-oriented standardization efforts • And much more Featuring over 50% new and revised material, this Second Edition of Setrag Khoshafian and Razmik Abnous's bestselling is now more than ever the best practical introduction to object technology for programmers and developers.

This volume contains the proceedings of the first European Conference on Object-Oriented Programming, held in Paris, June 15-17, 1987. The idea of this annual conference series is to provide a forum for theorists and practitioners interested in the object-oriented programming paradigm. The contributions cover the following aspects of object-oriented programming: methodology, implementation, Petri, interfaces, languages, simulation, inheritance.

Object-oriented Technology For Database And Software Systems

Software Maintenance - A Management Perspective

Journal of Object-oriented Programming

18th European Conference, Oslo, Norway, June 14-18, 2004, Proceedings

Library of Congress Subject Headings

A straightforward, step-by-step introduction to clear and elegant object-oriented programming. Using a language that's perfect for this kind of programming, the book has been tested in numerous courses and workshops over ten years. Programming Smalltalk is particularly suited for readers with no prior programming knowledge. Starting from the first principles of programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies. Not only is this book a step-by-step introduction to clear and elegant object-oriented programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies.

Quickly acquire the knowledge and skills you need to use object technology in your next development project A practical, down-to-earth introduction to object-oriented terms, concepts, and technologies, Object Orientation, Second Edition is for developers and programmers who are eager to start using object-oriented technology right away. Building step-by-step from the fundamentals to advanced design and development topics, this book supplies you with all the in-depth technical information and guidance you need to confidently incorporate object-oriented tools and techniques into your next project, no matter what your level of experience. Thanks to the authors' clear, straightforward explanations and professional insights, as well as the many real-world examples appearing throughout the book, you'll quickly acquire a solid working knowledge of • Abstract data typing, inheritance, and identity • Object-oriented analysis and design-including Booch, Rumbaugh, and other OOA and OOD methodologies • Object-oriented programming languages-including the object-oriented features of C++, Smalltalk, Ada, Eiffel, and other languages • Object-oriented database management systems-including OOODB, ORDB, client/server concepts, and examples from Object Design, Gem Stone, Versant, UniSQL, Objectivity, ODB-II, and other systems • Object-oriented GUI design-including explanations of Visual C++ and Foundation Classes, MacAPP, and NeXTStep • Object sharing and interchange with OLE 2 and OpenDoc • OMA, ODMG-93, and other object-oriented standardization efforts • And much more Featuring over 50% new and revised material, this Second Edition of Setrag Khoshafian and Razmik Abnous's bestselling is now more than ever the best practical introduction to object technology for programmers and developers.

This volume contains the proceedings of the first European Conference on Object-Oriented Programming, held in Paris, June 15-17, 1987. The idea of this annual conference series is to provide a forum for theorists and practitioners interested in the object-oriented programming paradigm. The contributions cover the following aspects of object-oriented programming: methodology, implementation, Petri, interfaces, languages, simulation, inheritance.

Object-oriented Technology For Database And Software Systems

Software Maintenance - A Management Perspective

Journal of Object-oriented Programming

18th European Conference, Oslo, Norway, June 14-18, 2004, Proceedings

Library of Congress Subject Headings

A straightforward, step-by-step introduction to clear and elegant object-oriented programming. Using a language that's perfect for this kind of programming, the book has been tested in numerous courses and workshops over ten years. Programming Smalltalk is particularly suited for readers with no prior programming knowledge. Starting from the first principles of programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies. Not only is this book a step-by-step introduction to clear and elegant object-oriented programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies.

Quickly acquire the knowledge and skills you need to use object technology in your next development project A practical, down-to-earth introduction to object-oriented terms, concepts, and technologies, Object Orientation, Second Edition is for developers and programmers who are eager to start using object-oriented technology right away. Building step-by-step from the fundamentals to advanced design and development topics, this book supplies you with all the in-depth technical information and guidance you need to confidently incorporate object-oriented tools and techniques into your next project, no matter what your level of experience. Thanks to the authors' clear, straightforward explanations and professional insights, as well as the many real-world examples appearing throughout the book, you'll quickly acquire a solid working knowledge of • Abstract data typing, inheritance, and identity • Object-oriented analysis and design-including Booch, Rumbaugh, and other OOA and OOD methodologies • Object-oriented programming languages-including the object-oriented features of C++, Smalltalk, Ada, Eiffel, and other languages • Object-oriented database management systems-including OOODB, ORDB, client/server concepts, and examples from Object Design, Gem Stone, Versant, UniSQL, Objectivity, ODB-II, and other systems • Object-oriented GUI design-including explanations of Visual C++ and Foundation Classes, MacAPP, and NeXTStep • Object sharing and interchange with OLE 2 and OpenDoc • OMA, ODMG-93, and other object-oriented standardization efforts • And much more Featuring over 50% new and revised material, this Second Edition of Setrag Khoshafian and Razmik Abnous's bestselling is now more than ever the best practical introduction to object technology for programmers and developers.

This volume contains the proceedings of the first European Conference on Object-Oriented Programming, held in Paris, June 15-17, 1987. The idea of this annual conference series is to provide a forum for theorists and practitioners interested in the object-oriented programming paradigm. The contributions cover the following aspects of object-oriented programming: methodology, implementation, Petri, interfaces, languages, simulation, inheritance.

Object-oriented Technology For Database And Software Systems

Software Maintenance - A Management Perspective

Journal of Object-oriented Programming

18th European Conference, Oslo, Norway, June 14-18, 2004, Proceedings

Library of Congress Subject Headings

A straightforward, step-by-step introduction to clear and elegant object-oriented programming. Using a language that's perfect for this kind of programming, the book has been tested in numerous courses and workshops over ten years. Programming Smalltalk is particularly suited for readers with no prior programming knowledge. Starting from the first principles of programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies. Not only is this book a step-by-step introduction to clear and elegant object-oriented programming, it teaches you how to use and create algorithms (reusable rules for problem-solving) and the basic building blocks of software. It goes on to explain how to develop complete applications and has a whole chapter on web applications as well as case studies.

Quickly acquire the knowledge and skills you need to use object technology in your next development project A practical, down-to-earth introduction to object-oriented terms, concepts, and technologies, Object Orientation, Second Edition is for developers and programmers who are eager to start using object-oriented technology right away. Building step-by-step from the fundamentals to advanced design and development topics, this book supplies you with all the in-depth technical information and guidance you need to confidently incorporate object-oriented tools and techniques into your next project, no matter what your level of experience. Thanks to the authors' clear, straightforward explanations and professional insights, as well as the many real-world examples appearing throughout the book, you'll quickly acquire a solid working knowledge of • Abstract data typing, inheritance, and identity • Object-oriented analysis and design-including Booch, Rumbaugh, and other OOA and OOD methodologies • Object-oriented programming languages-including the object-oriented features of C++, Smalltalk, Ada, Eiffel, and other languages • Object-oriented database management systems-including OOODB, ORDB, client/server concepts, and examples from Object Design, Gem Stone, Versant, UniSQL, Objectivity, ODB-II, and other systems • Object-oriented GUI design-including explanations of Visual C++ and Foundation Classes, MacAPP, and NeXTStep • Object sharing and interchange with OLE 2 and OpenDoc • OMA, ODMG-93, and other object-oriented standardization efforts • And much more Featuring over 50% new and revised material, this Second Edition of Setrag Khoshafian and Razmik Abnous's bestselling is now more than ever the best practical introduction to object technology for programmers and developers.

This volume contains the proceedings of the first European Conference on Object-Oriented Programming, held in Paris, June 15-17, 1987. The idea of this annual conference series is to provide a forum for theorists and practitioners interested in the object-oriented programming paradigm. The contributions cover the following aspects of object-oriented programming: methodology, implementation, Petri, interfaces, languages, simulation, inheritance.

Object-oriented Technology For Database And Software Systems