TCP/IP Sockets in Java: Practical Guide for Programmers

TCP/IP Sockets in Java: Practical Guide for Programmers is a quick and affordable way to gain the knowledge and skills needed to develop sophisticated and powerful web-based applications. The book's focused, tutorial-based approach enables the reader to master the tasks and techniques essential to virtually all client-server projects using sockets in Java. This edition has been expanded to include new advancements such as support for IPv6 as well as detailed defensive programming strategies. If you program using Java, be sure to check out this book's companion, TCP/IP Sockets in Java: Practical Guide for Programmers, Second Edition. Includes completely new and expanded sections that address the IPv6 network environment, defensive programming, and the select() system call, thereby allowing the reader to program in accordance with the most current standards for internetworking.

Streamlined and concise tutelage in conjunction with line-by-line code commentary allows readers to quickly program web-based applications without having to wade through unrelated and discursive networking tenets.

The networking capabilities of the Java platform have been extended considerably since the first edition of the book. This new edition covers version 1.5-1.7, the most current iterations, as well as making the following improvements:

- The API (application programming interface) reference sections in each chapter, which describe the relevant parts of each class, have been replaced with (i) a summary section that lists the classes and methods used in the code, and (ii) a “gotchas” section that mentions nonobvious or poorly-documented aspects of the objects.
- In addition, the book covers several new classes and capabilities introduced in the last few revisions of the Java platform. New abstractions to be covered include NetworkInterface, InterfaceAddress, Inet4/6Address, SocketAddress/InetSocketAddress, Executor, and others.
- Extended access to low-level network information.
- Support for IPv6.
- More complete access to socket options.
- Scalable I/O.
- The example code is also modified to take advantage of new language features such as annotations, enumerations, as well as generics and implicit iterators where appropriate.

Most Internet applications use sockets to implement network communication protocols. This book's focused, tutorial-based approach helps the reader master the tasks and techniques essential to virtually all client-server projects using sockets in Java. Chapter 1 provides a general overview of networking concepts to allow readers to synchronize the concepts with terminology. Chapter 2 introduces the mechanics of simple clients and servers. Chapter 3 covers basic message construction and parsing. Chapter 4 then deals with techniques used to build more robust clients and servers. Chapter 5 (NEW) introduces the scalable interface facilities which were introduced in Java 1.5, including the buffer and channel abstractions. Chapter 6 discusses the relationship between the programming constructs and the underlying protocol implementations in more detail.

Programming concepts are introduced through simple program examples accompanied by line-by-line code commentary.
Effective TCP/IP Programming offers valuable advice on such topics as:
- Exploring IP addressing, subnets, and CIDR
- Preferring the sockets interface over XTI/TLI
- Using two TCP connections
- Making your applications event-driven
- Using one large write instead of multiple small writes
- Avoiding data copying
- Understanding what TCP reliability really means
- Recognizing the effects of buffer sizes
- Using tcpdump, traceroute, netstat, and ping effectively

Numerous examples demonstrate essential ideas and concepts. Skeleton code and a library of common functions allow you to write applications without having to worry about routine chores. Through individual tips and explanations, you will acquire an overall understanding of TCP/IP's inner workings and the practical knowledge needed to put it to work. Using Effective TCP/IP Programming, you'll speed through the learning process and quickly achieve the programming capabilities of a seasoned pro.

A clear and comprehensive guide to TCP/IP protocols.

Introduction to JDBC

1. Presenting Information to Users
2. Querying the Database
3. Updating the Database
4. Advanced JDBC Topics
5. An eCommerce Example
6. How to Stay Current with JDBC
7. Appendix

SQL is a solid guide and reference to the key elements of SQL and how to use it effectively. Developed by authors who needed a good resource for students in their database class, this is an ideal supplement for database courses — no matter what main text you use or what flavor of SQL is required. It features a short and inexpensive introduction to SQL for students who have some programming experience and need to learn the main features of SQL; and suggested shortcuts for learning and practice, depending on the experience of the user. This book is recommended for novice developers, programmers, and database administrators as well as students in database courses, business courses, and IT-related courses. Provides tutorial-based instruction for the main features of SQL for programmers and other technical professionals in need of a brief but really good introduction to SQL. The approach is vendor-neutral—so very adaptable and flexible The focus is on teaching concepts by walking through concrete examples and explanations, and self-review exercises are included at the end of each chapter. Coverage is on the key features of the language that are required to understand SQL and begin using it effectively. SQL 2003-compliant.

Embedded Systems Architecture is a practical and technical guide to understanding the components that make up an embedded system's architecture. This book is perfect for those starting out as technical professionals such as engineers, programmers and designers of embedded systems; and also for students of computer science, computer engineering and
All of Java's Input/Output (I/O) facilities are based on streams, which provide simple ways to read and write data of different types. Java provides many different kinds of streams, each with its own application. The universe of streams is divided into four large categories: input streams and output streams, for reading and writing binary data; and readers and writers, for reading and writing textual (character) data. You're almost certainly familiar with the basic kinds of streams—but did you know that there's a CipherInputStream for reading encrypted data? And a ZipOutputStream for automatically compressing data? Do you know how to use buffered streams effectively to make your I/O operations more efficient?

Java I/O, 2nd Edition has been updated for Java 5.0 APIs and tells you all you ever need to know about streams—and probably more. A discussion of I/O wouldn't be complete without treatment of character sets and formatting. Java supports the Unicode standard, which provides definitions for the character sets of most written languages. Consequently, Java is the first programming language that lets you do I/O in virtually any language. Java also provides a sophisticated model for formatting textual and numeric data. Java I/O, 2nd Edition shows you how to control number formatting, use characters aside from the standard (but outdated) ASCII character set, and get a head start on writing truly multilingual software. Java I/O, 2nd Edition includes:

- Coverage of all I/O classes and related classes
- In-depth coverage of Java's number formatting facilities and its support for international character sets

This comprehensive resource, shows you everything you need to develop, compile, debug, and run Java programs. This expert guide has been updated for Java Platform Standard Edition 6 (Java SE 6) and offers complete coverage of the Java language, its syntax, keywords, and fundamental programming principles. Also find information on Java's key API libraries, learn to create applets and servlets, and use JavaBeans. The author has even included expanded coverage of Swing—the toolkit that defines the look and feel of the modern Java GUI.
TCP/IP Sockets in C# focuses on the Sockets API, the de facto standard for writing network applications in any programming language. Starting with simple client and server programs that use TCP/IP (the Internet protocol suite), students and practitioners quickly learn the basics and move on to firsthand experience with advanced topics including non-blocking sockets, multiplexing, threads, asynchronous programming, and multicasting. Key network programming concepts such as framing, performance and deadlocks are illustrated through hands-on examples. Using a detailed yet clear, concise approach, this book includes numerous code examples and focused discussions to provide a solid understanding of programming TCP/IP sockets in C#.

Features:
- Tutorial-based instruction in key sockets programming techniques complemented by numerous code examples throughout.
- Discussion moves quickly into the C# Sockets API definition and code examples, desirable for those who want to get up-to-speed quickly.
- Important coverage of "under the hood" details that developers will find useful when creating and using a socket or a higher level TCP class that utilizes sockets.
- Includes end-of-chapter exercises to facilitate learning, as well as sample code available for download at the book's companion web site.

Dive into key topics in network architecture and Go, such as data serialization, application level protocols, character sets and encodings. This book covers network architecture and gives an overview of the Go language as a primer, covering the latest Go release. Beyond the fundamentals, Network Programming with Go covers key networking and security issues such as HTTP and HTTPS, templates, remote procedure call (RPC), web sockets including HTML5 web sockets, and more. Additionally, author Jan Newmarch guides you in building and connecting to a complete web server based on Go. This book can serve as both an essential learning guide and reference on Go networking.
This book is for experienced Go programmers and other programmers with some experience with the Go language. Harness the hidden power of Java to build network-enabled applications with lower network traffic and faster processes.

About This Book
Learn to deliver superior server-to-server communication through the networking channels. Gain expertise of the networking features of your own applications to support various network architectures such as client/server and peer-to-peer. Explore the issues that impact scalability, affect security, and allow applications to work in a heterogeneous environment.

Who This Book Is For
Learning Network Programming with Java is oriented to developers who wish to use network technologies to enhance the utility of their applications. You should have a working knowledge of Java and an interest in learning the latest in network programming techniques using Java. No prior experience with network development or special software beyond the Java SDK is needed. Upon completion of the book, beginner and experienced developers will be able to use Java to access resources across a network and the Internet.

What You Will Learn
Connect to other applications using sockets. Use channels and buffers to enhance communication between applications. Access network services and develop client/server applications. Explore the critical elements of peer-to-peer applications and current technologies available. Use UDP to perform multicasting. Address scalability through the use of core and advanced threading techniques. Incorporate techniques into an application to make it more secure. Configure and address interoperability issues to enable your applications to work in a heterogeneous environment.

In Detail
Network-aware applications are becoming more prevalent and play an ever-increasing role in the world today. Connecting and using an Internet-based service is a frequent requirement for many applications. Java provides numerous classes that have evolved over the years to meet evolving network needs. These range from low-level socket and IP-based approaches to those encapsulated in software services. This book explores how Java supports networks, starting with the basics and then advancing to more complex topics. An overview of each relevant network technology is presented followed by detailed examples of how to use Java to support these technologies. We start with the basics of networking and then explore how Java supports the development of client/server and peer-to-peer applications. The NIO packages are examined as well as multitasking and how network applications can address practical issues such as security. A discussion on networking concepts will put many network issues into perspective and let you focus on the appropriate technology for the problem at hand. The examples used will provide a good starting point to develop similar capabilities for many of your network needs.

Style and approach
Each network technology's terms and concepts are introduced first. This is followed up with code examples to explain these technologies. Many of the examples are supplemented with alternate Java 8 solutions when appropriate. Knowledge of Java 8 is not necessary but these examples will help you better understand the power of Java 8. Java's rich, comprehensive networking interfaces make it an ideal platform for building today's networked, Internet-centered applications, components, and Web services. Now, two Java networking experts demystify Java's complex networking API, giving developers practical insight into the key techniques of network development, and providing extensive code examples that show exactly how it's done.

David and Michael Reilly begin by reviewing fundamental Internet architecture and TCP/IP protocol concepts all network programmers need to understand, as well as general Java features and techniques that are especially important in network programming, such as exception handling and input/output. Using practical examples, they show how to write clients and servers using UDP and TCP; how to build multithreaded network applications; and how to utilize HTTP and access the Web using Java. The book includes detailed coverage of server-side application development; distributed computing development with RMI and CORBA; and email-enabling applications with the powerful JavaMail API. For all beginning to intermediate Java programmers, network programmers who need to learn to work with Java.

How prepared are you to build fast and efficient web applications? This eloquent book provides what every web developer should know about the network, from fundamental limitations that affect performance to major innovations for building even more powerful browser applications—including HTTP 2.0 and XHR improvements, Server-Sent Events (SSE), WebSocket, and WebRTC. Author Ilya Grigorik, a web performance engineer at Google, demonstrates
TCP/IP Sockets in Java: Practical Guide for Programmers, with its focused, tutorial-based coverage, helps you master the tasks and techniques essential to virtually all client-server projects using sockets in Java. Later...
TCP/IP Sockets in Java Practical Guide For Programmers

The Practical Guides chapters teach you to implement more specialized functionality; incisive discussions of programming constructs and protocol implementations equip you with a deeper understanding that is invaluable for meeting future challenges. No other resource presents so concisely or so effectively the exact material you need to get up and running with Java sockets programming right away. For those who program using the C language, be sure to check out this book's companion, TCP/IP Sockets in C: Practical Guide for Programmers. For example code from the text, sample programming exercises, Powerpoint slides, and more, click on the grey “Companion Site” button to the right. *Concise, no-nonsense explanations of issues often troublesome for students, including message construction and parsing, underlying mechanisms and Java I/O *Comprehensive example-based coverage of the most important TCP/IP techniques—including iterative and threaded servers, timeouts and asynchronous message processing *Includes a detailed, easy-to-use reference to the relevant JAVA class libraries *A companion Web site provides online code for all the example programs given in the book *Provides a guide to common errors and a reference offering detailed documentation of the sockets interface *Perfect for a practitioner who may even want just to “look into” this technology. *Provides tutorial-based instruction in key sockets programming techniques, focusing exclusively on Java and complemented by example code. *Covers challenging sockets programming issues: message construction and parsing, underlying TCP/IP protocol mechanisms, Java I/O, iterative and threaded servers, and timeouts. *Includes references to the relevant Java class libraries that often go beyond the “official” Java documentation in clarity and explanation. *Provides code for all example programs, along with additional exercises, via companion Web site. Multicast Sockets: Practical Guide for Programmers is a hands-on, application-centric approach to multicasting (as opposed to a network-centric one) that is filled with examples, ideas, and experimentation. Each example builds on the last to introduce multicast concepts, frameworks, and APIs in an engaging manner that does not burden the reader with lots of theory and jargon. The book is an introduction to multicasting but assumes that the reader has a background in network programming and is proficient in C or Java. After reading the book, you will have a firm grasp on how to write a multicast program. Author team of instructor and application programmer is reflected in this rich instructional and practical approach to the subject material Only book available that provides a clear, concise, application-centric approach to programming multicast applications and covers several languages—C, Java, and C# on the .NET platform Covers important topics like service models, testing reachability, and addressing and scoping Includes numerous examples and exercises for programmers and students to test what they have learned The Java New I/O (NIO) packages in J2SE 1.4 introduce many new, indispensable features previously unavailable to Java programmers. These include APIs for high-performance I/O operations, regular expression processing, and character set coding. These new libraries are a treasure trove for Java developers. The NIO APIs are especially valuable where high-performance I/O is a requirement, but they can also be useful in a wide range of scenarios. The new APIs let you work directly with I/O buffers, multiplex nonblocking streams, do scattering reads and gathering writes, do channel-to-channel transfers, work with memory-mapped files, manage file locks, and much more. The new high-performance Regular Expression Library provides sophisticated, Perl-like regex-processing features such as pattern matching, search and replace, capture groups, look ahead assertions, and many others. The Charset API gives you complete control over character set encoding and decoding, which are vital for properly managing the exchange of documents on the Web, for localization, or for other purposes. You can also create and install your own custom character sets. Staying current with the latent Java technology is never easy. NIO, new in Java 1.4, is quite possibly the most important new Java feature since Swing. Understanding it thoroughly is essential for any serious Java developer. NIO closes the gap between Java and natively compiled languages and enables Java applications to achieve maximum I/O performance by effectively leveraging operating-system services in a portable way. Java NIO is a comprehensive guide to the Java New I/O facilities. It lets you take full advantage of NIO features and shows you how they work, what they can do for you, and when you should use them. This book brings you up to speed on NIO and shows you how to bring your I/O-bound Java applications up to speed as well. Java NIO is an essential part of any Java professional’s library.
systems. This second edition of this bestselling guide is updated to reflect the Servlet API 2.2, how to effectively deploy a servlet-based application, security and user authentication, and explain the new JSP technology and new information on databases and JDBC. The CD-ROM includes an updated sample servlet code.

TCP/IP Sockets in C: Practical Guide for Programmers, Second Edition is a quick and affordable way to gain the knowledge and skills needed to develop sophisticated and powerful web-based applications. The book's focused, tutorial-based approach enables the reader to master the tasks and techniques essential to virtually all client-server projects using sockets in C. This edition has been expanded to include new advancements such as support for IPv6 as well as detailed defensive programming strategies. If you program using Java, be sure to check out this book's companion, TCP/IP Sockets in Java: Practical Guide for Programmers, 2nd Edition. Includes completely new and expanded sections that address the IPv6 network environment, defensive programming, and the select() system call, thereby allowing the reader to program in accordance with the most current standards for internetworking.

Streamlined and concise tutelage in conjunction with line-by-line code commentary allows readers to quickly program web-based applications without having to wade through unrelated and discursive networking tenets. This volume focuses on the underlying sockets class, one of the basis for learning about networks in any programming language. By learning to write simple client and server programs that use TCP/IP, readers can then realize network routing, framing, error detection and correction, and performance.

A guide to developing network programs covers networking fundamentals as well as TCP and UDP sockets, multicasting protocol, content handlers, servlets, I/O, parsing, Java Mail API, and Java Secure Sockets Extension. From Charles M. Kozierok, the creator of the highly regarded www.pcguide.com, comes The TCP/IP Guide. This completely up-to-date, encyclopedic reference on the TCP/IP protocol suite will appeal to newcomers and the seasoned professional alike. Kozierok details the core protocols that make TCP/IP internetworks function and the most important classic TCP/IP applications, integrating IPv6 coverage throughout. Over 350 illustrations and hundreds of tables help to explain the finer points of this complex topic. The book's personal, user-friendly writing style lets readers of all levels understand the dozens of protocols and technologies that run the Internet, with full coverage of PPP, ARP, IP, IPv6, IP NAT, IPSec, Mobile IP, ICMP, RIP, BGP, TCP, UDP, DNS, DHCP, SNMP, FTP, SMTP, NNTP, HTTP, Telnet, and much more. The TCP/IP Guide is a must-have addition to the libraries of internetworking students, educators, networking professionals, and those working toward certification.

IPv6 was introduced in 1994 and has been in development at the IETF for over 10 years. It has now reached the deployment stage. KAME, the de-facto open-source reference implementation of the IPv6 standards, played a significant role in the acceptance and the adoption of the IPv6 technology. The adoption of KAME by key companies in a wide spectrum of commercial products is a testimonial to the success of the KAME project, which concluded not long ago. This book is the first and the only one of its kind, which reveals all of the details of the KAME IPv6 protocol stack, explaining exactly what every line of code does and why it was designed that way. Through the dissection of both the code and its design, the authors illustrate how IPv6 and its related protocols have been interpreted and implemented from the specifications. This reference will demystify those ambiguous areas in the standards, which are open to interpretation and problematic in deployment, and presents solutions offered by KAME in dealing with these implementation challenges. Covering a snapshot version of KAME dated April 2003 based on FreeBSD 4.8, extensive line-by-line code listings with meticulous explanation of their rationale and use for the KAME snapshot implementation, which is generally applicable to most recent versions of the KAME IPv6 stack including those in recent releases of BSD variants. Numerous diagrams and illustrations help in visualizing the implementation. In-depth discussion of the standards provides intrinsic understanding of the specifications.

JXTA: Java P2P Programming provides an invaluable introduction to this new technology, filled with useful information and practical examples. It was created by members of the JXTA community, sharing their real-world experience to introduce developers to JXTA. It starts with the fundamentals of P2P and demonstrates how JXTA fulfills the P2P promise, then covers the essentials of JXTA including the protocols, the JXTA Shell, and groups. Later chapters include case studies demonstrating
Acces PDF Tcp Ip Sockets In Java Practical Guide For Programmers The Practical Guides

The Practical Guides

JXTA to synchronize data and to create distributed applications. Includes a forward by Juan Carlos Soto, Group Marketing Manager for Project JXTA at Sun Microsystems and the jxta.org Open Source Community Manager. A package which provides an in-depth tutorial on programming networked applications with Java. It offers complete coverage of the Java networking APIs, including streams, TCP/IP and UDP/IP, with practical examples. The package presents a cryptographic framework for developing Internet applications. This title focuses on using Java for building network computing solutions. The CD-ROM includes sample code from the book and free software, including VisualAge for Java, Lotus Bean Machine, Servlet Express, Web Runner Bean Tools, and Lotus Domino Go Web server. The 1st edition of this book was equally useful as an undergraduate textbook and as the lucid, no-nonsense guide required by IT professionals, featuring many code examples, screenshots and exercises. The new 2nd edition adds revised language reflecting significant changes in J2SE 5.0; update of support software; non-blocking servers; DataSource interface and Data Access Objects for connecting to remote databases.* Clear and abundant examples, using real-world code, written by three experienced developers who write networking code for a living. * Describes how to build clients and servers, explains how TCP, UDP, and IP work, and shows how to debug networking applications via packet sniffing and deconstruction. * Well suited for Windows developer looking to expand to Linux, or for the proficient Linux developer looking to incorporate client-server programming into their application.

The networking capabilities of the Java platform have been extended considerably since the first edition of the book. This new edition covers version 1.5-1.7, the most current iterations, as well as making the following improvements: The API (application programming interface) reference sections in each chapter, which describe the relevant parts of each class, have been replaced with (i) a summary section that lists the classes and methods used in the code, and (ii) a “gotchas” section that mentions nonobvious or poorly-documented aspects of the objects. In addition, the book covers several new classes and capabilities introduced in the last few revisions of the Java platform. New abstractions to be covered include NetworkInterface, InterfaceAddress, Inet4/6Address, SocketAddress/InetSocketAddress, Executor, and others; extended access to low-level network information; support for IPv6; more complete access to socket options; and scalable I/O. The example code is also modified to take advantage of new language features such as annotations, enumerations, as well as generics and implicit iterators where appropriate. Most Internet applications use sockets to implement network communication protocols. This book's focused, tutorial-based approach helps the reader master the tasks and techniques essential to virtually all client-server projects using sockets in Java. Chapter 1 provides a general overview of networking concepts to allow readers to synchronize the concepts with terminology. Chapter 2 introduces the mechanics of simple clients and servers. Chapter 3 covers basic message construction and parsing. Chapter 4 then deals with techniques used to build more robust clients and servers. Chapter 5 (NEW) introduces the scalable interface facilities which were introduced in Java 1.5, including the buffer and channel abstractions. Chapter 6 discusses the relationship between the programming constructs and the underlying protocol implementations in more detail. Programming concepts are introduced through simple program examples accompanied by line-by-line code commentary that describes the purpose of every part of the program. No other resource presents so concisely or so effectively the material necessary to get up and running with Java sockets programming. Focused, tutorial-based instruction in key sockets programming techniques allows reader to quickly come up to speed on Java applications. Concise and up-to-date coverage of the most recent platform (1.7) for Java applications in networking technology. Helps readers eliminate performance problems, covering topics including bottlenecks, profiling tools, strings, algorithms, distributed systems, and servlets. A tutorial introducing Java basics covers programming principles, integrating applets with Web applications, and using threads, arrays, and sockets.

Copyright code : 0b27acb6e4a56b705523d2e81c748e4e

Copyright : ns1.thepixel.com

Page 9/9