

《Java 网络编程核心技术详解》

第 1 章 Java 网络编程入门

- 1. 1 进程之间的通信
- 1. 2 计算机网络的概念
- 1. 3 OSI 参考模型
- 1. 4 TCP/IP 参考模型和 TCP/IP
 - 1. 4. 1 IP
 - 1. 4. 2 TCP 以及端口
 - 1. 4. 3 RFC 简介
 - 1. 4. 4 客户/服务器通信模式
- 1. 5 用 Java 编写客户/服务器程序
 - 1. 5. 1 创建 EchoServer
 - 1. 5. 2 创建 EchoClient
- 1. 6 小结
- 1. 7 练习题

第 2 章 Socket 用法详解

- 2. 1 构造 Socket
 - 2. 1. 1 设定等待建立连接的超时时间
 - 2. 1. 2 设定服务器的地址
 - 2. 1. 3 设定客户端的地址
 - 2. 1. 4 客户连接服务器时可能抛出的异常
 - 2. 1. 5 使用代理服务器
 - 2. 1. 6 InetAddress 地址类的用法
 - 2. 1. 7 NetworkInterface 类的用法
- 2. 2 获取 Socket 的信息
- 2. 3 关闭 Socket
- 2. 4 半关闭 Socket
- 2. 5 设置 Socket 的选项
 - 2. 5. 1 TCP_NODELAY 选项
 - 2. 5. 2 SO_REUSEADDR 选项
 - 2. 5. 3 SO_TIMEOUT 选项
 - 2. 5. 4 SO_LINGER 选项
 - 2. 5. 5 SO_RCVBUF 选项
 - 2. 5. 6 SO_SNDBUF 选项
 - 2. 5. 7 SO_KEEPALIVE 选项
 - 2. 5. 8 OOBINLINE 选项
 - 2. 5. 9 IP 服务类型选项
 - 2. 5. 10 设定连接时间、延迟和带宽的相对重要性
- 2. 6 发送邮件的 SMTP 客户程序
- 2. 7 小结
- 2. 8 练习题

第 3 章 ServerSocket 用法详解

3.1 构造 ServerSocket

3.1.1 绑定端口

3.1.2 设定客户连接请求队列的长度

3.1.3 设定绑定的 IP 地址

3.1.4 默认构造方法的作用

3.2 接收和关闭与客户的连接

3.3 关闭 ServerSocket

3.4 获取 ServerSocket 的信息

3.5 ServerSocket 选项

3.5.1 SO_TIMEOUT 选项

3.5.2 SO_REUSEADDR 选项

3.5.3 SO_RCVBUF 选项

3.5.4 设定连接时间、延迟和带宽的相对重要

3.6 创建多线程的服务器

3.6.1 为每个客户分配一个线程

3.6.2 创建线程池

3.6.3 使用 JDK 类库提供的线程池

3.6.4 向线程池提交有异步运算结果的任务

3.6.5 使用线程池的注意事项

3.7 关闭服务器

3.8 小结

3.9 练习题

第 4 章 非阻塞通信

4.1 线程阻塞的概念

4.1.1 线程阻塞的原因

4.1.2 服务器程序用多线程处理阻塞通信的局限

4.1.3 非阻塞通信的基本思想

4.2 非阻塞通信 API 的用法

4.2.1 缓冲区

4.2.2 字符编码 Charset

4.2.3 通道

4.2.4 SelectableChannel 类

4.2.5 ServerSocketChannel 类

4.2.6 SocketChannel 类

4.2.7 Selector 类

4.2.8 SelectionKey 类

4.2.9 Channels 类

4.2.10 Socket 选项

4.3 服务器编程范例

4.3.1 创建阻塞的 EchoServer

4.3.2 创建非阻塞的 EchoServer

4.3.3 在 EchoServer 中混合用阻塞模式与非阻塞模式

4.4 客户端编程范例

- 4. 4. 1 创建阻塞的 `EchoClient`
- 4. 4. 2 创建非阻塞的 `EchoClient`
- 4. 5 异步通道和异步运算结果
- 4. 6 在 GUI 中用 `SwingWorker` 实现异步交互
- 4. 6. 1 `SwingWorker` 类的用法
- 4. 6. 2 用 `SwingWorker` 类来展示进度条
- 4. 6. 3 用 `SwingWorker` 类实现异步的 `AsynEchoClient`
- 4. 7 小结
- 4. 8 练习题
- 第 5 章 创建非阻塞的 HTTP 服务器
- 5. 1 HTTP 简介
- 5. 1. 1 HTTP 请求格式
- 5. 1. 2 HTTP 响应格式
- 5. 1. 3 测试 HTTP 请求
- 5. 2 创建非阻塞的 HTTP 服务器
- 5. 2. 1 服务器主程序: `HttpServer` 类
- 5. 2. 2 具有自动增长的缓冲区的 `ChannelIO` 类
- 5. 2. 3 负责处理各种事件的 `Handler` 接口
- 5. 2. 4 负责处理接收连接就绪事件的 `AcceptHandler` 类
- 5. 2. 5 负责接收 HTTP 请求和发送 HTTP 响应的 `RequestHandler` 类
- 5. 2. 6 代表 HTTP 请求的 `Request` 类
- 5. 2. 7 代表 HTTP 响应的 `Response` 类
- 5. 2. 8 代表响应正文的 `Content` 接口及其实现类
- 5. 2. 9 运行 HTTP 服务器
- 5. 3 小结
- 5. 4 练习题
- 第 6 章 客户端协议处理框架
- 6. 1 客户端协议处理框架的主要类
- 6. 2 在客户程序中运用协议处理框架
- 6. 2. 1 `URL` 类的用法
- 6. 2. 2 `URLConnection` 类的用法
- 6. 3 实现协议处理框架
- 6. 3. 1 创建 `EchoURLConnection` 类
- 6. 3. 2 创建 `EchoURLStreamHandler` 及工厂类
- 6. 3. 3 创建 `EchoContentHandler` 类及工厂类
- 6. 3. 4 在 `EchoClient` 类中运用 ECHO 协议处理框架
- 6. 4 小结
- 6. 5 练习题
- 第 7 章 用 Swing 组件展示 HTML 文档
- 7. 1 在按钮等组件上展示 HTML 文档
- 7. 2 用 `JEditorPane` 组件创建简单的浏览器
- 7. 2. 1 处理 HTML 页面上的超级链接
- 7. 2. 2 处理 HTML 页面上的表单
- 7. 2. 3 创建浏览器程序

- 7. 3 小结
- 7. 4 练习题
- 第 8 章 基于 UDP 的数据报和套接字
- 8. 1 UDP 简介
- 8. 2 DatagramPacket 类
 - 8. 2. 1 选择数据报的大小
 - 8. 2. 2 读取和设置 DatagramPacket 的属性
 - 8. 2. 3 数据格式的转换
 - 8. 2. 4 重用 DatagramPacket
- 8. 3 DatagramSocket 类
 - 8. 3. 1 构造 DatagramSocket
 - 8. 3. 2 接收和发送数据报
 - 8. 3. 3 管理连接
 - 8. 3. 4 关闭 DatagramSocket
 - 8. 3. 5 DatagramSocket 的选项
 - 8. 3. 6 IP 服务类型选项
- 8. 4 DatagramChannel 类
 - 8. 4. 1 创建 DatagramChannel
 - 8. 4. 2 管理连接
 - 8. 4. 3 用 send()方法发送数据报
 - 8. 4. 4 用 receive()方法接收数据报
 - 8. 4. 5 用 write()方法发送数据报
 - 8. 4. 6 用 read()方法接收数据报
 - 8. 4. 7 Socket 选项
- 8. 5 组播
 - 8. 5. 1 MulticastSocket 类
 - 8. 5. 2 组播 Socket 的范例
- 8. 6 小结
- 8. 7 练习题
- 第 9 章 对象的序列化与反序列化
- 9. 1 JDK 类库中的序列化 API
 - 9. 1. 1 把对象序列化到文件
 - 9. 1. 2 把对象序列化到网络
- 9. 2 实现 Serializable 接口
 - 9. 2. 1 序列化对象图
 - 9. 2. 2 控制序列化的行为
 - 9. 2. 3 readResolve()方法在单例类中的运用
- 9. 3 实现 Externalizable 接口
- 9. 4 可序列化类的不同版本的序列化兼容性
- 9. 5 小结
- 9. 6 练习题
- 第 10 章 Java 语言的反射机制
- 10. 1 Java Reflection API 简介
- 10. 2 在远程方法调用中运用反射机制

- 10. 3 代理模式
 - 10. 3. 1 静态代理类
 - 10. 3. 2 动态代理类
 - 10. 3. 3 在远程方法调用中运用代理类
- 10. 4 小结
- 10. 5 练习题
- 第 11 章 RMI 框架
 - 11. 1 RMI 的基本原理
 - 11. 2 创建第 1 个 RMI 应用
 - 11. 2. 1 创建远程接口
 - 11. 2. 2 创建远程类
 - 11. 2. 3 创建服务器程序
 - 11. 2. 4 创建客户程序
 - 11. 2. 5 运行 RMI 应用
 - 11. 3 远程对象工厂设计模式
 - 11. 4 远程方法中的参数与返回值传递
 - 11. 5 回调客户端的远程对象
 - 11. 6 远程对象的并发访问
 - 11. 7 分布式垃圾收集
 - 11. 8 远程对象的 equals()、hashCode()和 clone()方法
 - 11. 9 使用安全管理器
 - 11. 10 RMI 应用的部署以及类的动态加载
 - 11. 11 远程激活
 - 11. 12 小结
 - 11. 13 练习题
- 第 12 章 通过 JDBC API 访问数据库
 - 12. 1 JDBC 的实现原理
 - 12. 2 安装和配置 MySQL 数据库
 - 12. 3 JDBC API 简介
 - 12. 4 JDBC API 的基本用法
 - 12. 4. 1 处理字符编码的转换
 - 12. 4. 2 把连接数据库的各种属性放在配置文件中
 - 12. 4. 3 管理 Connection、Statement 和 ResultSet 对象的生命周期
 - 12. 4. 4 执行 SQL 脚本文件
 - 12. 4. 5 处理 SQLException
 - 12. 4. 6 输出 JDBC 日志
 - 12. 4. 7 获得新插入记录的主键值
 - 12. 4. 8 设置批量抓取属性
 - 12. 4. 9 检测驱动器使用的 JDBC 版本
 - 12. 4. 10 元数据
 - 12. 5 可滚动以及可更新的结果集
 - 12. 6 行集
 - 12. 7 调用存储过程
 - 12. 8 处理 Blob 和 Clob 类型数据

- 12. 9 控制事务
 - 12. 9. 1 事务的概念
 - 12. 9. 2 声明事务边界的概念
 - 12. 9. 3 在 `mysql. exe` 程序中声明事务
 - 12. 9. 4 通过 JDBC API 声明事务边界
 - 12. 9. 5 保存点
 - 12. 9. 6 批量更新
 - 12. 9. 7 设置事务隔离级别
- 12. 10 数据库连接池
 - 12. 10. 1 创建连接池
 - 12. 10. 2 DataSource 数据源
- 12. 11 小结
- 12. 12 练习题
- 第 13 章 基于 MVC 和 RMI 的分布式应用
 - 13. 1 MVC 设计模式简介
 - 13. 2 store 应用简介
 - 13. 3 创建视图
 - 13. 4 创建控制器
 - 13. 5 创建模型
 - 13. 6 创建独立应用
 - 13. 7 创建分布式应用
 - 13. 8 小结
 - 13. 9 练习题
- 第 14 章 通过 JavaMail API 收发邮件
 - 14. 1 E-mail 协议简介
 - 14. 1. 1 SMTP
 - 14. 1. 2 POP3
 - 14. 1. 3 接收邮件的新协议 IMAP
 - 14. 1. 4 MIME 简介.
 - 14. 2 JavaMail API 简介
 - 14. 3 建立 JavaMail 应用程序的开发环境
 - 14. 3. 1 获得 JavaMail API 的类库
 - 14. 3. 2 安装和配置邮件服务器
 - 14. 4 创建 JavaMail 应用程序
 - 14. 5 身份验证
 - 14. 6 授权码验证
 - 14. 7 URLName 类
 - 14. 8 创建和读取复杂电子邮件
 - 14. 8. 1 邮件地址
 - 14. 8. 2 邮件头部
 - 14. 8. 3 邮件标记
 - 14. 8. 4 邮件正文
 - 14. 9 操纵邮件夹
 - 14. 10 小结

- 14. 11 练习题
- 第 15 章 安全网络通信
 - 15. 1 SSL 简介
 - 15. 1. 1 加密通信
 - 15. 1. 2 安全证书
 - 15. 1. 3 SSL 握手
 - 15. 1. 4 创建自我签名的安全证书
 - 15. 2 JSSE 简介
 - 15. 2. 1 KeyStore、KeyManager 与 TrustManager 类
 - 15. 2. 2 SSLContext 类
 - 15. 2. 3 SSLServerSocketFactory 类
 - 15. 2. 4 SSLSocketFactory 类
 - 15. 2. 5 SSLSocket 类
 - 15. 2. 6 SSLServerSocket 类
 - 15. 2. 7 SSLEngine 类
 - 15. 3 创建基于 SSL 的安全服务器和安全客户
 - 15. 4 小结
 - 15. 5 练习题
- 第 16 章 XML 数据处理
 - 16. 1 用 DOM 处理 XML 文档
 - 16. 2 用 SAX 处理 XML 文档
 - 16. 2. 1 创建 XML 文档的具体处理类 CustomerHandler
 - 16. 2. 2 创建 XML 文档的解析类 SaxDemo
 - 16. 3 用 JDOM 处理 XML 文档
 - 16. 4 用 DOM4J 处理 XML 文档
 - 16. 5 Java 对象的 XML 序列化和反序列化
 - 16. 6 小结
 - 16. 7 练习题
- 第 17 章 用 Axis 发布 Web 服务
 - 17. 1 SOAP 简介
 - 17. 2 建立 Apache Axis 环境
 - 17. 3 在 Tomcat 上发布 Apache-Axis Web 应用
 - 17. 4 创建 SOAP 服务
 - 17. 4. 1 创建提供 SOAP 服务的 Java 类
 - 17. 4. 2 创建 SOAP 服务的发布描述文件
 - 17. 5 发布和管理 SOAP 服务
 - 17. 5. 1 发布 SOAP 服务
 - 17. 5. 2 管理 SOAP 服务
 - 17. 6 创建和运行 SOAP 客户程序
 - 17. 7 小结
 - 17. 8 练习题
- 第 18 章 用 Spring 整合 CXF 发布 Web 服务
 - 18. 1 创建 Web 服务接口和实现类
 - 18. 2 在 Spring 配置文件中配置 Web 服务

18. 3 在 web.xml 配置文件中配置 Spring 和 CXF

18. 4 在 Tomcat 中发布 Web 服务

18. 5 创建和运行客户程序

18. 6 小结

18. 7 练习题

附录 A 本书范例的运行方法

A. 1 本书所用软件的下载地址

A. 2 部分软件的安装

A. 2. 1 安装 JDK

A. 2. 2 安装 ANT

A. 2. 3 安装 Tomcat

A. 3 编译源程序

A. 4 运行客户/服务器程序

A. 5 处理编译和运行错误