

图书基本信息

书名：<<Tomcat与Java Web开发技术详解>>

13位ISBN编号：9787302177241

10位ISBN编号：7302177244

出版时间：2008-6

出版时间：清华大学出版社

作者：封超，晁阳 编著

页数：475

字数：722000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

内容概要

利用Java语言不仅能开发桌面版应用程序，JavaEE更是广泛应用于基于B/S架构的Web企业应用程序领域。

尤其是EJB、JSP、Servlet、JDBC等技术使得用Java语言编写的企业级应用程序更易于维护和扩展。

本书共19章，分为三篇。

第一篇为基础篇，从第1章到第11章，详细介绍了Java语言的基础语法和理论知识等内容，为后续两篇的学习打下了坚实的基础；第二篇为Web编程篇，从第12章到第16章，分别讲解了利用Java语言开发Web应用程序所要用到的各项技术，包括JSP技术、Servlet技术、JavaBean技术以及Java数据库编程技术（JDBC、SQL基础和MySQL数据库的使用），还介绍Tomcat容器的安装配置和使用方法，特别是第15章，要求读者认真阅读和理解；第三篇为实例篇，从第17章到第19章，深入讲解了MVC设计模式、在线相册案例和网上商品案例，此篇为前两个篇章的综合应用篇。

本书通过作者简洁详实的语言和思路清晰的知识结构为读者学习Java语言以及如何开发Web应用程序做了深入的剖析。

本书适合作为初学者的指导书，也适合有一定Java语言基础的读者使用。

书籍目录

| | | | |
|----------------------|---------------------------------|---------------------------------|------------------------------------|
| 第一篇 基础篇 | 第1章 Java语言概述 | 1.1 Java语言的发展与特点 | 1.1.1 Java语言的发展 |
| 1.1.2 Java语言的特点 | 1.1.3 Java语言的执行 | 1.2 JDK的安装和配置 | 1.2.1 在Microsoft Windows平台上安装配置JDK |
| 编写HelloWorld | 1.2.2 在Linux平台上安装配置JDK | 1.3 Java程序举例 | 1.3.1 在Linux平台上编写和运行HelloWorld |
| 运行HelloWorld | 1.3.2 编译Java源文件 | 1.3.3 运行Java程序 | 1.3.4 在Linux平台上编写和运行HelloWorld |
| 1.4.3 注释 | 1.4 Java语言编码规范 | 1.4.1 Java源文件以及文件组织 | 1.4.2 缩进排版 |
| 1.5.1 JCreator简介 | 1.4.4 声明 | 1.4.5 空白行 | 1.4.6 命名规范 |
| 标识符和关键字 | 1.5.2 安装和使用JCreator | 1.6 本章小结 | 第2章 Java编程基础 |
| 字符 | 2.1.1 标识符 | 2.1.2 Java中的保留关键字 | 2.1.3 注释、分隔符及转义 |
| 2.3.3 定义方法 | 2.2 基本数据类型及其常量值 | 2.3 类 | 2.3.1 什么是类 |
| 2.5 包和main方法 | 2.3.4 变量的声明 | 2.4 对象 | 2.3.2 类的声明和修饰符 |
| 本章小结 | 2.4.1 创建对象 | 2.4.2 this关键字 | 2.6 本章小结 |
| 3.1.3 比较运算符 | 2.5.1 包的声明和导入 | 2.5.2 程序入口main ()方法的声明 | 第3章 运算符与表达式 |
| 反运算符 | 3.1 运算符 | 3.1.1 赋值运算符 | 3.1.2 算术运算符 |
| 数据类型 | 3.1.3 比较运算符 | 3.1.4 移位运算符 | 3.1.5 等于运算符 |
| 建 | 3.1.8 逻辑运算符 | 3.2 给方法传递参数 | 3.1.6 位运算符 |
| 5.1 分支语句 | 3.3 本章小结 | 4.1 数组简介 | 3.2.1 原始数据类型 |
| 句 | 4.1.3 初始化数组 | 4.2 java.util.Arrays类 | 3.2.2 变量 |
| 关键字 | 5.1.1 if-else语句 | 4.3 本章小结 | 第4章 数组 |
| 再抛出 | 5.2.2 do-while循环语句 | 5.2 循环语句 | 4.1.1 数组的声明 |
| 6.1.2 封装的应用 | 5.2.3 for循环语句 | 5.2.1 while循环语句 | 4.1.2 数组的初始化 |
| 7.1 构造函数 | 5.3 标号语句 | 5.2.4 在循环语句中使用break和continue | 第5章 流程控制和异常处理 |
| 7.2.1 重写规则 | 5.4 异常处理 | 5.4.1 异常的层次结构 | 5.4.2 try-catch处理异常 |
| 声明和实现 | 5.4.3 使用finally | 5.4.4 异常的匹配 | 5.4.5 编写并使用自己的异常类 |
| 7.3.4 接口的多重继承 | 5.4.5 编写并使用自己的异常类 | 5.4.6 异常的finally | 5.5 本章小结 |
| 7.4.3 封装类的常用方法 | 5.5 本章小结 | 第6章 面向对象程序设计 | 6.1 封装及其应用 |
| 8.1.1 字符与字符串 | 6.1.2 封装的应用 | 6.2 继承 | 6.1.1 封装的特点 |
| 法 | 7.1 构造函数 | 6.3 多态 | 6.2 继承 |
| StringBuffer类的主要方法 | 7.2.1 重写规则 | 6.4 本章小结 | 6.3 多态 |
| 8.3.3 综合举例 | 7.2.2 方法的重写 | 第7章 构造函数、重载和重写、接口 | 6.4 本章小结 |
| ()方法 | 7.3.1 使用接口实现契约式设计 | 7.1 构造函数 | 7.1.1 构造函数基础 |
| 9.2 集合框架 | 7.3.4 接口的多重继承 | 7.1.2 构造函数的重载 | 7.1.2 构造函数的重载 |
| Set接口及其实现类 | 7.4.3 封装类的常用方法 | 7.2.3 重载规则 | 7.2 重写和重载方法 |
| 接口及其实现类 | 8.1.1 字符与字符串 | 7.2.4 方法的重载 | 7.2.4 方法的重载 |
| 及其实现类 | 8.1.2 String类 | 7.3 接口的声明和实现 | 7.3 接口的声明和实现 |
| TreeMap类 | 8.1.3 字符串对象的创建 | 7.3.1 使用接口实现契约式设计 | 7.3.2 定义接口 |
| 10.1.2 内部类的实例化与访问 | 8.2 StringBuffer类及其方法 | 7.3.3 实现接口 | 7.3.3 实现接口 |
| 10.5 本章小结 | 8.2.1 StringBuffer类 | 7.4 封装类 | 7.4.1 封装类简介 |
| 程 | 8.2.2 append ()方法 | 7.4.1 封装类简介 | 7.4.2 创建封装器类对象 |
| 级 | 8.2.3 StringBuffer类的主要方法 | 7.4.2 创建封装器类对象 | 7.4.2 创建封装器类对象 |
| 第二篇 Web编程篇 | 8.3 Pattern类和Matcher类 | 7.4.3 封装类的常用方法 | 7.5 本章小结 |
| 装Tomcat | 8.3.1 Pattern类 | 7.5 本章小结 | 第8章 字符串的处理 |
| 12.3.3 测试JSP | 8.3.2 Matcher类 | 8.1 String类及其重要方法 | 8.1 String类及其重要方法 |
| 12.3.4 测试Servlet | 8.3.3 综合举例 | 8.1.1 字符与字符串 | 8.1.1 字符与字符串 |
| 12.4 Tomcat 6.0的目录结构 | 8.4 Java数学运算 | 8.1.2 String类 | 8.1.2 String类 |
| 12.5 配置Tomcat 5.5 | 8.4.1 java.lang.Math类 | 8.1.3 字符串对象的创建 | 8.1.3 字符串对象的创建 |
| | 8.4.2 java.math.BigInteger类 | 8.1.4 String类的主要方法 | 8.1.4 String类的主要方法 |
| | 8.4.3 java.math.BigDecimal类 | 8.2 StringBuffer类及其方法 | 8.2 StringBuffer类及其方法 |
| | 8.5 本章小结 | 8.2.1 StringBuffer类 | 8.2.1 StringBuffer类 |
| | 第9章 集合框架 | 8.2.2 append ()方法 | 8.2.2 append ()方法 |
| | 9.1 重写hashCode ()和equals ()方法 | 8.2.3 StringBuffer类的主要方法 | 8.2.3 StringBuffer类的主要方法 |
| | 9.1.1 Object类 | 8.3 Pattern类和Matcher类 | 8.3 Pattern类和Matcher类 |
| | 9.1.2 重写equals ()方法 | 8.3.1 Pattern类 | 8.3.1 Pattern类 |
| | 9.1.3 重写hashCode ()方法 | 8.3.2 Matcher类 | 8.3.2 Matcher类 |
| | 9.2 集合框架 | 8.3.3 综合举例 | 8.3.3 综合举例 |
| | 9.2.1 Java集合框架简介 | 8.4 Java数学运算 | 8.4 Java数学运算 |
| | 9.2.2 集合框架中常用的接口及其实现类 | 8.4.1 java.lang.Math类 | 8.4.1 java.lang.Math类 |
| | 9.3 Set接口及其实现类 | 8.4.2 java.math.BigInteger类 | 8.4.2 java.math.BigInteger类 |
| | 9.3.1 HashSet类 | 8.4.3 java.math.BigDecimal类 | 8.4.3 java.math.BigDecimal类 |
| | 9.3.2 LinkedHashSet类 | 8.5 本章小结 | 8.5 本章小结 |
| | 9.3.3 TreeSet类 | 第9章 集合框架 | 第9章 集合框架 |
| | 9.4 List接口及其实现类 | 9.1 重写hashCode ()和equals ()方法 | 9.1 重写hashCode ()和equals ()方法 |
| | 9.4.1 LinkedList类 | 9.1.1 Object类 | 9.1.1 Object类 |
| | 9.4.2 ArrayList类 | 9.1.2 重写equals ()方法 | 9.1.2 重写equals ()方法 |
| | 9.4.3 Vector类 | 9.1.3 重写hashCode ()方法 | 9.1.3 重写hashCode ()方法 |
| | 9.5 Map接口及其实现类 | 9.2 集合框架 | 9.2 集合框架 |
| | 9.5.1 Hashtable类 | 9.2.1 Java集合框架简介 | 9.2.1 Java集合框架简介 |
| | 9.5.2 LinkedHashMap类 | 9.2.2 集合框架中常用的接口及其实现类 | 9.2.2 集合框架中常用的接口及其实现类 |
| | 9.5.3 HashMap类 | 9.3 Set接口及其实现类 | 9.3 Set接口及其实现类 |
| | 9.5.4 TreeMap类 | 9.3.1 HashSet类 | 9.3.1 HashSet类 |
| | 9.6 本章小结 | 9.3.2 LinkedHashSet类 | 9.3.2 LinkedHashSet类 |
| | 第10章 内部类 | 9.3.3 TreeSet类 | 9.3.3 TreeSet类 |
| | 10.1 内部类简介 | 9.4 List接口及其实现类 | 9.4 List接口及其实现类 |
| | 10.1.1 编写普通的内部类 | 9.4.1 LinkedList类 | 9.4.1 LinkedList类 |
| | 10.1.2 内部类的实例化与访问 | 9.4.2 ArrayList类 | 9.4.2 ArrayList类 |
| | 10.2 局部内部类 | 9.4.3 Vector类 | 9.4.3 Vector类 |
| | 10.3 匿名内部类 | 9.5 Map接口及其实现类 | 9.5 Map接口及其实现类 |
| | 10.4 顶级嵌套类 | 9.5.1 Hashtable类 | 9.5.1 Hashtable类 |
| | 10.5 本章小结 | 9.5.2 LinkedHashMap类 | 9.5.2 LinkedHashMap类 |
| | 第11章 多线程 | 9.5.3 HashMap类 | 9.5.3 HashMap类 |
| | 11.1 多线程基础 | 9.5.4 TreeMap类 | 9.5.4 TreeMap类 |
| | 11.1.1 线程的定义 | 9.6 本章小结 | 9.6 本章小结 |
| | 11.1.2 实例化线程 | 第10章 内部类 | 第10章 内部类 |
| | 11.2 线程的阻塞和优先级 | 10.1 内部类简介 | 10.1 内部类简介 |
| | 11.2.1 线程睡眠 | 10.1.1 编写普通的内部类 | 10.1.1 编写普通的内部类 |
| | 11.2.2 线程的优先级 | 10.1.2 内部类的实例化与访问 | 10.1.2 内部类的实例化与访问 |
| | 11.3 同步 | 10.2 局部内部类 | 10.2 局部内部类 |
| | 11.3.1 同步与锁定 | 10.3 匿名内部类 | 10.3 匿名内部类 |
| | 11.3.2 死锁 | 10.4 顶级嵌套类 | 10.4 顶级嵌套类 |
| | 11.4 本章小结 | 10.5 本章小结 | 10.5 本章小结 |
| | 第12章 Tomcat服务器简介 | 11.1 多线程基础 | 11.1 多线程基础 |
| | 12.1 Tomcat简介 | 11.1.1 线程的定义 | 11.1.1 线程的定义 |
| | 12.2 下载Tomcat | 11.1.2 实例化线程 | 11.1.2 实例化线程 |
| | 12.3 安装Tomcat | 11.2 线程的阻塞和优先级 | 11.2 线程的阻塞和优先级 |
| | 12.3.1 Tomcat在Windows平台的安装 | 11.2.1 线程睡眠 | 11.2.1 线程睡眠 |
| | 12.3.2 Tomcat在Linux平台的安装 | 11.2.2 线程的优先级 | 11.2.2 线程的优先级 |
| | 12.3.3 测试JSP | 11.3 同步 | 11.3 同步 |
| | 12.3.4 测试Servlet | 11.3.1 同步与锁定 | 11.3.1 同步与锁定 |
| | 12.4 Tomcat 6.0的目录结构 | 11.3.2 死锁 | 11.3.2 死锁 |
| | 12.5 配置Tomcat 5.5 | 11.4 本章小结 | 11.4 本章小结 |

<<Tomcat与Java Web开发 >

| | | | |
|----------------------------------|----------------------------------|--------------------------------|-----------------------|
| Administration Web Application | 12.6 Java开源Web框架简介 | 12.6.1 WebWork | 12.6.2 Struts |
| 12.6.3 Spring | 12.6.4 Java Server Faces (JSF) | 12.6.5 Tapestry | 12.7 本章小结 |
| 第12章 JSP编程 | 13.1 JSP概述 | 13.1.1 JSP简介 | 13.1.2 一个简单的JSP程序 |
| 13.2 JSP基本语法 | 13.2.1 JSP语法 | 13.2.2 操作Cookie | 13.2 JSP基本语法 |
| 13.3.1 request对象 | 13.3.2 response对象 | 13.3.3 out对象 | 13.3.4 session对象 |
| 13.3.5 application对象 | 13.3.6 config对象 | 13.3.7 pageContext对象 | 13.3.8 page对象 |
| 13.3.9 exception对象 | 13.4 本章小结 | 第14章 Servlet编程 | 14.1 Servlet简介 |
| 14.1.1 Servlet的主要功能和特点 | 14.1.2 Servlet的基本结构 | 14.2 Servlet的生命周期 | 14.3 Servlet编程类和接口 |
| 14.3.1 | 14.3.2 HttpServletRequest接口 | 14.3.3 HttpServletResponse接口 | 14.4 JSP |
| HttpSession接口 | 14.5 本章小结 | 第15章 Java数据库编程 | 15.1 安装和配置MySQL数据库 |
| 与Servlet | 15.1.2 下载MySQL数据库 | 15.1.3 安装和配置MySQL | 15.2 数据库基础 |
| MySQL数据库简介 | 15.2.1 SQL简介 | 15.2.2 数据查询语言 | 15.2.3 数据控制语言 (DCL) |
| —SQL编程 | 15.2.4 数据定义语言 (DDL) | 15.2.5 数据操纵语言 (DML) | 15.3 通过JDBC访问MySQL数 |
| 据库 | 15.3.1 JDBC简介 | 15.3.2 获取MySQL数据库支持JAR包 | 15.3.3 JDBC访问MySQL |
| 数据库 | 15.4 数据源 (DataSource) 简介 | 15.5 在Tomcat 6.0中配置MySQL数据源连接池 | 15.6 |
| 在Java程序中访问数据源 | 15.6.1 JNDI基础 | 15.6.2 Java程序访问数据源 | 15.7 处理MySQL |
| 中文编码 | 15.8 SQLJ编程 | 15.8.1 SQLJ简介 | 15.8.2 SQLJ与JDBC |
| 15.8.3 使用SQLJ | 15.9 本章小结 | 第16章 JavaBean编程 | 16.1 JavaBean简介 |
| 问Oracle数据库 | 16.2.1 如何编写JavaBean | 16.2.2 在JSP中使用JavaBean | 16.2 JavaBean规范 |
| 16.3 | 16.2.3 JavaBean与EJB | 16.3 | 16.3 |
| 本章小结 | 第三篇 实例篇 | 第17章 MVC设计模式 | 17.1 MVC设计模式简介 |
| 17.2 SmartUpload组 | 17.2.1 SmartUpload组件基础 | 17.2.2 编写前台JSP | 17.2.3 编写中心控制器Servlet |
| 17.2.4 使用SmartUpload组件下载文件 | 17.2.5 在Tomcat中部署并运行应用程序 | 17.3 | 17.3 |
| JSP+Servlet+JavaBean综合案例——用户登录验证 | 17.3.1 数据库实现 | 17.3.2 开发持久 | 17.3.3 开发中心控制器Servlet |
| 层JavaBean | 17.3.4 开发显示层JSP | 17.3.5 在Tomcat中部署并运行应用程序 | 17.4 本章小结 |
| 17.4 本章小结 | 第18章 网上商店综合案例 | 18.1 系统分析 | 18.1.1 概 |
| 18.1.2 需求分析 | 18.2 系统架构设计 | 18.2.1 前台架构分析 | 18.2.2 后台架构分析 |
| 18.3 数据库设计 | 18.3.1 MySQL数据类型参考 | 18.3.2 数据库建表 | 18.3.3 数据库表关 |
| 系图 | 18.4 开发数据库JavaBean | 18.5 系统前台主要功能模块的设计与实现 | 18.5.1 用户注册 |
| 模块 | 18.5.2 用户登录模块 | 18.5.3 商品搜索模块 | 18.5.4 用户管理模块 |
| 18.5.5 | 18.5.6 购物车模块 | 18.6 系统后台主要功能的模块设计与实现 | 18.6.1 |
| 户资料修改模块 | 18.6.2 商品管理模块 | 18.6.3 商品图片更新 | 18.6.4 处理用户订单 |
| 商品搜索模块 | 18.7 在Tomcat中部署并运行应用程序 | 18.7.1 web.xml文件的配置 | 18.7.2 应用程序的目录 |
| 18.7.3 运行网上商店应用程序 | 18.8 本章小结 | 第19章 在线相册综合案例 | 19 |
| 系统设计 | 19.1.1 概述 | 19.1.2 需求分析 | 19.2 系统架构设计 |
| 19.2.1 系统架构分析 | 19.2 功能分析说明 | 19.3 数据库设计 | 19.3.1 MySQL存取图片 |
| 19.3.2 数据库建表 | 19.4 开发数据库JavaBean | 19.5 系统主要功能模块的设计与实现 | 19.5.1 用户身份验证模块 |
| 19.5.2 照片显示模块 | 19.5.3 用户上传照片模块 | 19.5.4 新建相册模块 | 19.5.5 删除 |
| 相册模块 | 19.6 在Tomcat中部署并运行应用程序 | 19.6.1 web.xml文件和数据库源的配置 | 19.6.2 应用程序的目录结构设计 |
| 19.6.3 运行在线相册应用程序 | 19.7 本章小结 | | |

章节摘录

第1章 Java语言概述 本章主要介绍Java语言的发展、特点，JDK的下载和安装以及Java语言编码规范等。

Java是由Sun Microsystems公司于1995年5月推出的，其中包括：Java程序设计语言（以下简称Java）和Java运行平台（包括Java虚拟机和Java应用编程接口（Java Application Programming Interface，API））。

Java语言相对于C、C++等编程语言具有更好的安全性，Java是跨平台的、面向对象的、简单的、面向网络的编程语言，这也是Java得以快速发展的根本所在。

网络技术给现代企业带来了许多新的机遇和挑战，如何改善客户关系服务、全球化和信息检索等成为现代企业面对的主要难题，这些在技术上反映为信息的获取、系统管理、系统集成、新技术的开发、Intranet等与商业的结合。

Java语言的出现迅速引起所有程序员和软件公司的极大关注，他们纷纷尝试用Java语言编写网络应用程序，并利用网络把程序发布到世界各地进行运行。

包括IBM、Oracle、微软、SGI等公司纷纷与Sun Microsystems公司签订合同，被授权使用Java平台技术。

目前，Java语言已经成为最流行的网络编程语言，截止到2001年，全世界大约有310万Java程序员，许多高等院校纷纷开设Java课程，Java正逐步成为世界上使用人数最多的编程语言。

微软公司总裁比尔·盖茨先生曾经说过：“Java语言是长时间以来最卓越的程序设计语言”。

编辑推荐

《Tomcat与Java Web开发技术详解》通过作者简洁详实的语言和思路清晰的知识结构为读者学习Java语言以及如何开发Web应用程序做了深入的剖析。

《Tomcat与Java Web开发技术详解》适合作为初学者的指导书，也适合有一定Java语言基础的读者使用。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>