第一图书网, tushu007.com

<<JAVASCRIPT语言精髓与编程实 >

图书基本信息

书名：<<JAVASCRIPT语言精髓与编程实践>>

13位ISBN编号：9787121056871

10位ISBN编号：7121056879

出版时间：2008-3

出版时间：电子工业出版社

作者：周爱民

页数：510

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

本书详细讲述JavaScript作为一种混合式语言的各方面特性，包括过程式、面向对象、函数式和动态语言特性等，在动态函数式语言特性方面有着尤为细致的讲述。
本书的主要努力之一，就是分解出这些语言原子，并重现将它们混合在一起的过程与方法。
通过从复杂性到单一语言特性的还原过程，读者可了解到语言的本质，以及“层出不穷的语言特性”背后的真相。
本书主要的著述目的是基于一种形式上简单的语言来讲述“语言的本质及其应用”。
本书详细讲述了通过框架执行过程来构造一个JavaScript扩展框架的方法，并完整地讲述了框架扩展中各种设计取舍，因此可以作为学习研究计算机程序设计语言时的参考，用以展示现实系统如何实现经典理论中的各种编程范型。

作者简介

周爱民，国内软件开发界资深软件工程师，从1996年开始涉足商业软件开发，历任部门经理、区域总经理、高级软件工程师、平台架构师等职。
周爱民先生在软件开发、软件工程、团队建设以及部门管理方面经验丰富，是Borland Delphi产品技术专家，也是Qomo开源项目（JavaScript）的发起者。
在JavaScript开发方面，有超过9年的实践经验。

2003年5月，被美国Borland公司授予“ Borland Delphi产品专家 ”称号，并授予“ 论坛特别贡献奖 ”。

2004年8月，出版软件开发专著《Delphi源代码分析》，被业界誉为“ Delphi领域精品著作 ”。

2005年12月，发起JavaScript开源项目Qomolangma（Qomo），研究语言特性、框架设计与企业级应用库。

2007年4月，出版软件工程专著《大道至简——软件工程实践者的思想》，被誉为“ 激荡新思的佳作 ”。

2007年11月，参加CSDN主办的软件开发大会（SD2C），主讲“ JavaScript引擎技术 ”。

书籍目录

第1部分 语言基础 第1章 十年JAVASCRIPT 1.1 网页中的代码 1.1.1 新鲜的玩意儿 1.1.2 第一段在网页中的代码 1.1.3 最初的价值 1.2 用JAVASCRIPT来写浏览器上的应用 1.2.1 我要做一个聊天室 1.2.2 Flash的一席之地 1.2.3 RWC与RIA之争 1.3 没有框架与库的语言能怎样发展呢？ 1.3.1 做一个框架 1.3.2 重写框架的语言层 1.3.3 富浏览器端开发（RWC）与AJAX 1.4 为JAVASCRIPT正名 1.4.1 JavaScript 1.4.2 Core JavaScript 1.4.3 SpiderMonkey JavaScript 1.4.4 ECMAScript 1.4.5 JScript 1.4.6 总述 1.5 JAVASCRIPT的应用环境 1.5.1 宿主环境（host environment） 1.5.2 外壳程序（Shell） 1.5.3 运行期环境（runtime） 第2章 JAVASCRIPT的语法 2.1 语法综述 2.1.1 标识符所绑定的语义 2.1.2 识别语法错误与运行错误 2.2 JAVASCRIPT的语法：变量声明 2.2.1 变量的数据类型 2.2.2 变量声明 2.2.3 变量声明中的一般性问题 2.3 JAVASCRIPT的语法：表达式运算 2.3.1 一般表达式运算 2.3.2 逻辑运算 2.3.3 字符串运算 2.3.4 比较运算 2.3.5 赋值运算 2.3.6 函数调用 2.3.7 特殊作用的运算符 2.3.8 运算优先级 2.4 JAVASCRIPT的语法：语句 2.4.1 表达式语句 2.4.2 分支语句 2.4.3 循环语句 2.4.4 流程控制：一般子句 2.4.5 流程控制：异常 2.5 面向对象编程的语法概要 2.5.1 对象直接量声明与实例创建 2.5.2 对象成员列举、存取和删除 2.5.3 属性存取与方法调用 2.5.4 对象及其成员的检查 2.5.5 可列举性 2.5.6 默认对象的指定 2.6 运算符的二义性 2.6.1 加号“+”的二义性 2.6.2 括号“()”的二义性 2.6.3 冒号“：”与标签的二义性 2.6.4 大括号“{}”的二义性 2.6.5 逗号“,”的二义性 2.6.6 方括号“[]”的二义性 第2部分 语言特性及基本应用 第3章 JAVASCRIPT的非函数式语言特性 3.1 概述 3.1.1 命令式语言与结构化编程 3.1.2 结构化的疑难 3.1.3 “面向对象语言”是突破吗？ 3.1.4 更高层次的抽象：接口 3.1.5 再论语言的分类 3.1.6 JavaScript的语源 3.2 基本语法的结构化含义 3.2.1 基本逻辑与代码分块 3.2.2 模块化的层次：语法作用域 3.2.3 执行流程及其变更 3.2.4 模块化的效果：变量作用域 3.2.5 语句的副作用 3.3 JAVASCRIPT中的原型继承 3.3.1 空对象（null）与空的对象 3.3.2 原型继承的基本性质 3.3.3 空的对象是所有对象的基础 3.3.4 构造复制？
写时复制？
还是读遍历？
3.3.5 构造过程：从函数到构造器 3.3.6 预定义属性与方法 3.3.7 原型链的维护 3.3.8 原型继承的实质 3.4 JAVASCRIPT的对象系统 3.4.1 封装 3.4.2 多态 3.4.3 事件 3.4.4 类抄写？
或原型继承？
3.4.5 JavaScript中的对象（构造器） 3.4.6 不能通过继承得到的效果 第4章 JAVASCRIPT的函数式语言特性 4.1 概述 4.1.1 从代码风格说起 4.1.2 为什么常见的语言不赞同连续求值 4.1.3 函数式语言的渊源 4.2 函数式语言中的函数 4.2.1 函数是运算元 4.2.2 在函数内保存数据 4.2.3 函数内的运算对函数外无副作用 4.3 从运算式语言到函数式语言 4.3.1 JavaScript中的几种连续运算 4.3.2 运算式语言 4.3.3 如何消灭掉语句 4.4 函数：对运算式语言的补充和组织 4.4.1 函数是必要的补充 4.4.2 函数是代码的组织形式 4.4.3 重新认识“函数” 4.4.4 JavaScript语言中的函数式编程 4.5 JAVASCRIPT中的函数 4.5.1 可变参数与值参数传递 4.5.2 非惰性求值 4.5.3 函数是第一型 4.5.4 函数是一个值 4.5.5 可遍历的调用栈 4.6 闭包 4.6.1 什么是闭包 4.6.2 什么是函数实例与函数引用 4.6.3 （在被调用时，）每个函数实例至少拥有一个闭包 4.6.4 函数闭包与调用对象 4.6.5 函数实例拥有多个闭包的情况 4.6.6 语句或语句块中的闭包问题 4.6.7 闭包中的标识符（变量）特例

4.6.8 函数对象的闭包及其效果 4.6.9 闭包与可见性 第5章 JAVASCRIPT的动态语言特性 5.1 概述 5.1.1 动态数据类型的起源 5.1.2 动态执行系统的起源 5.1.3 脚本系统的起源 5.1.4 脚本只是一种表面的表现形式 5.2 动态执行（EVAL） 5.2.1 动态执行与闭包 5.2.2 动态执行过程中的语句、表达式与值 5.2.3 奇特的、甚至是负面的影响 5.3 动态方法调用（CALL与APPLY） 5.3.1 动态方法调用中指定this对象 5.3.2 丢失的this引用 5.3.3 栈的可见与修改 5.3.4 兼容性：低版本中的call()与apply() 5.4 重写 5.4.1 原型重写 5.4.2 构造器重写 5.4.3 对象成员的重写 5.4.4 宿主对重写的限制 5.4.5 引擎对重写的限制 5.5 包装类：面向对象的妥协 5.5.1 显式包装元数据 5.5.2 隐式包装的过程与检测方法 5.5.3 包装值类型数据的必要性与问题 5.5.4 其他直接量与相应的构造器 5.6 关联数组：对象与数组的动态特性 5.6.1 关联数组是对象系统的基础 5.6.2 用关联数组实现的索引数组 5.6.3 干净的对象 5.7 类型转换 5.7.1 宿主环境下的特殊类型系统 5.7.2 值运算：类型转换的基础 5.7.3 隐式转换 5.7.4 值类型之间的转换 5.7.5 从引用到值：深入探究valueOf()方法 5.7.6 到字符串类型的显式转换 第3部分 编程实践 第6章 QOMO框架的核心技术与实现 6.1 QOMO框架的技术发展与基本特性 6.1.1 Qomo框架的技术发展 6.1.2 Qomo的体系结构 6.1.3 Qomo框架设计的基本原则 6.2 基于类继承的对象系统 6.2.1 Qomo类继承的基本特性 6.2.2 Qomo类继承的语法 6.2.3 Qomo类继承系统的实现 6.2.4 Qomo类继承系统的高级话题 6.3 多投事件系统 6.3.1 多投事件系统的基本特性与语法 6.3.2 多投事件系统的实现 6.3.3 多投事件的中断与返回值 6.3.4 多投事件系统的安全性 6.4 接口系统 6.4.1 基本概念与语法 6.4.2 接口实现 6.4.3 Qomo接口系统的高级话题 6.4.4 接口委托 6.4.5 Qomo接口系统的实现 6.5 命名空间 6.5.1 Qomo命名空间的复杂性 6.5.2 命名空间的使用 6.5.3 命名空间的实现 6.6 AOP 6.6.1 基本概念与语法 6.6.2 高级切面特性 6.6.3 Qomo中切面系统的实现 6.7 其他 6.7.1 装载、内联与导入 6.7.2 四种模板技术 6.7.3 出错处理 6.7.4 其他功能模块 第7章 一般性的动态函数式语言技巧 7.1 消除代码的全局变量名占用 7.2 一次性的构造器 7.3 对象充当识别器 7.4 识别NEW运算进行的构造器调用 7.5 使用直接量及其包装类快速调用对象方法 7.6 三天前是星期几？ 7.7 使用对象的值含义来构造复杂对象 7.8 控制字符串替换过程的基本模式 7.9 实现二叉树 7.10 将函数封装为方法 7.11 使用WITH语句来替代函数参数传递 7.12 使用对象闭包来重置重写 7.13 构造函数参数 7.14 使用更复杂的表达式来消减IF语句 7.15 利用钩子函数来扩展功能 7.16 安全的字符串 附录A：术语表 附录B：主要引擎的特性差异列表 附录C：附图 附录D：参考书目

章节摘录

插图：

媒体关注与评论

“这是一本闪烁着思考光芒的作品，它的精彩就在于作者深入的思考。”
——蒋涛
“停下来，思考才是进步的本质。”
——李维
“传达了探究工程本质的独立思考精神。”
——谭群钊

编辑推荐

《JAVASCRIPT语言精髓与编程实践》适合于有编程经验的、想学习和掌握JavaScript的语言实现与扩展知识的开发者。

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