第一图书网, tushu007.com

<<Everyday Scripting w>>

图书基本信息

书名：<<Everyday Scripting with Ruby中文版>>

13位ISBN编号：9787121064579

10位ISBN编号：712106457X

出版时间：2008-5

出版时间：电子工业出版社

作者：马瑞克

页数：301

字数：374000

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

这是一本关于Ruby特性和技巧的书，特性会在项目需要用到的时候进行介绍，每个部分也会在以前技巧的基础上引入新的技巧。
本书将向读者介绍Ruby的一些基础知识，如何使用测试驱动设计来稳定可控地编写脚本，如何使用应用程序框架，如何处理常见错误，等等。
　　书中包含大量精彩的诀窍、指导、警告、案例、隐喻、练习以及提问，对于那些想最大限度地利用测试自动化的软件测试从业者、分析师、程序员来说，本书极具参考价值。

书籍目录

第1章　引言　1.1　关于本书结构　1.2　全书梗概　1.3　售后服务　1.4　续篇　1.5　鸣谢第2章　起步　2.1　下载练习文件　2.2　从命令行开始　2.3　你需要安装Ruby吗　2.4　安装Ruby　2.5　两个基本工具　2.6　命令行提示符和irb提示符　2.7　是犯错的时候了第1部分　基础知识　第3章　第一个脚本：比较文件清单　3.1　一个实战脚本　3.2　Ruby世界　3.3　对象发送和接收消息　3.4　用变量来命名对象　3.5　比较数组　3.6　打印到屏幕　3.7　制作一个脚本　3.8　我们进行到哪了　3.9　练习　第4章　Ruby语言细节：数组　第5章　三处改进和一处缺陷修正　5.1　命令行参数　5.2　忽略大小写　5.3　方法　5.4　分割字符串　5.5　修正一个Bug　5.6　我们在哪儿　5.7　练习的前奏　5.8　练习　第6章　Ruby语言细节：If、相等性判断和Unless　6.1　if…elsif…else　6.2　什么时候对象相等　6.3　if的一个快捷方式　6.4　unless　6.5　问号操作符第Ⅱ部分　一步一步编写脚本　第7章　Chutn项目：轻松编写脚本　7.1　演示项目　7.2　构建解决方案　7.3　我们进行到哪了　7.4　练习　第8章　Ruby语言细节：布尔型　8.1　其他布尔运算符　8.2　优先级　8.3　所有对象皆为真值　8.4　通过布尔表达式选择对象　第9章　利器在手，正则表达式　9.1　用正则表达式匹配字符串　9.2　用正则表达式解析字符串　9.3　重新排列数组　9.4　我们进行到哪了　9.5　练习　第10章　Ruby语言细节：正则表达式　10.1　特殊字符　10.2　分组和多选结构　10.3　拆分字符串　10.4　幕后的变量　10.5　正则表达式选项　10.6　等等，还没完呢　10.7　练习　第11章　用类捆绑数据和方法　11.1　在类里定义方法　11.2　用对象包含数据　11.3　我们进行到哪了　11.4　练习　第12章　Ruby语言细节：类（附带介绍Symbol）　12.1　定义访问方法　12.2　Self　12.3　类方法　12.4　类变量和全局变量　12.5　练习第Ⅲ部分　在活生生的世界中工作　第13章　用正则表达式刮取网页　13.1　像处理文件一样处理网页　13.2　把注意力限定在网页的有用部分　13.3　抽取标题和作者　13.4　用散列表存储有名数据　13.5　开始亲缘图书旅程　13.6　亲自练习　第14章　其他使用Web应用的方法　14.1　处理XHTML　14.2　驱动浏览器　14.3　直接访问底层协议　第15章　处理逗号分隔的数据　15.1　CSV库　15.2　使用代码块进行自动清理　15.3　更多CSV操作　15.4　把所有这些应用到affinity-trip.rb中　15.5　发现并理解标准库中的类　15.6　用数据取代代码　第16章　Ruby语言细节：散列表　第17章　Ruby语言细节：参数列表　17.1　可选参数　17.2　剩余参数　17.3　关键字参数　第18章　下载辅助脚本和应用　18.1　寻找包　18.2　使用setup.rb　18.3　使用RubyGems　18.4　理解你所下载的　第19章　打磨你的脚本　19.1　加载路径　19.2　避免文件名冲突　19.3　使用模块避免类命名冲突　19.4　让脚本替你完成工作　19.5　轻松自如的工作　19.6　rakefile　19.7　位置无关的测试　19.8　练习　第20章　Ruby语言细节：模块　20.1　嵌套模块　20.2　包含模块　20.3　类是模块　第21章　当脚本出现问题的时候　21.1　使用异常报告问题　21.2　一种错误处理策略　21.3　异常处理选项　21.4　使用Block的方法　21.5　练习第Ⅳ部分　成为行家　第22章　框架：用填空法编写脚本　22.1　使用watchdog脚本　22.2　继承　22.3　收集用户选择　第23章　发现比创建更安全　23.1　Barker的故事　23.2　发生了什么？
在哪儿发生？
23.3　用模块替代超类　第24章最后总结第Ⅴ部分　书后附录　附录A　词汇表　附录B　练习答案　B.1　第3章练习答案　B.2　第5章练习答案　B.3　第7章练习答案　B.4　第9章练习答案　B.5　第10章练习答案　B.6　第11章练习答案　B.7　第12章练习答案　B.8　第21章练习答案　附录C　参考文献索引

章节摘录

第1章　引言　　鞋匠的孩子光着脚到处跑。
那些不从事软件开发的人会发现，大量的工具不断涌现，使得工作负担不断从人类转移到计算机。
但是软件开发人员常常有不同的观点。
他们日复一日地重复那些手工劳动：在一张工作台前，一个测试人员正在用手工方式把测试数据输入到数据库里；在另外一张工作台，一个程序员正筛查版本控制系统的输出结果，试图找到想要的文件；在第三张工作台，一个商业分析师正将报表里的数据拷贝到一个电子表格中。
以上这些工作计算机可以做得很完美，人们为什么要自己做?这与知识和技能有关。
测试人员认为编程太难，因此从来不学；程序员会编程，但是他所会的语言无法轻松完成此项工作，并且他没有时间利用颇有难度的方式来实现这些任务：那位分析师曾写过一个脚本来完成类似的杂事，但是当他试着把脚本应用到那个报表时，脚本却不能工作，使这个脚本重新工作会比手工拷贝数据花费更多时间，即使他下个月还会拷贝6次。
本书是为下面这几类人写的：　　为那些认为编程太难的人（我们的测试人员）：其实编程没有那么难。
编程的名声不太好，是因为计算机曾经运行得很慢。
为了使程序运行得足够快，程序员不得不使用那些必须告诉计算机各种各样繁琐细节的编程语言。
现在计算机已经足够快了，能使用那些不需要阐述所有细节的语言。
结果就是，现在编程简单多了。
为那些在编写或改写较大脚本时陷入困境的人（我们的分析师）：你还不能掌控复杂性，在本书中你能学到这些。
本书会教你一种现代风格的编程方式，它强调测试先行（测试驱动的编程），尽可能借鉴别人的工作，逐步扩展程序，并且让代码一直保持干净整洁。
很多脚本是一次性的：编写、使用、然后扔到一边。
不过对于那些希望保留的脚本，这些技能会助你一臂之力。
（说实话，我接触过的许多专业程序员也没有学到这些技能，因此他们会发现这本书是一个不错的开始。
）　　为那些熟知“错误”的编程语言的人（我们的程序员）：象Java、C#、c++或者C这样的编程语言有特定的使用环境，但是它们并不善于快速实现小程序，尤其是那些需要操作文本和文件而非数字和内部数据结构的小程序。
你需要在自己的技能清单里添上另一门编程语言。
在本书里，你将学习一种对上述三类读者都很适合的语言——RubY。
它易学易用，既有转换或者搜索文本这类简单的特性，也具备处理复杂事物的各种特性。
如果你是测试人员，你会很高兴，因为测试也是Ruby所擅长的领域之一（很大程度上归功于Watir）。
如果你是程序员，你大概已经知道Ruby最近因为它的“杀手级应用”——Rails（一个构建Web应用的框架）而变得爆受欢迎。
抛开这些，Ruby已有超过l0年的历史，它不是某种短暂的时髦东西或者不稳定的原型。
而且，Ruby社区十分友好，每个人都会乐于参与其中。

媒体关注与评论

大量精彩的诀窍、指导、警告、全面的例子、隐喻、练习以及提问！
对于那些想最大限度地利用测试自动化的软件测试从业者来说，本书极具参考价值。
GrigoriMelnik UniversityofCalgary，讲师 一本精彩的“跟我一起做（type-along-with-me）”类型的书籍，它由浅入深地介绍了一门强大的脚本语言，让读者在不知不觉中成为一名Ruby程序员。
ErikPetersen Emprove 终于，测试社区拥有了一本饱含智慧结晶的手册。
文字简明易懂，示例代码面向现实问题，并涵盖像测试驱动、继承这样的复杂主题。
通过它，测试人员不仅能在测试领域更上层楼，也让他们能为所在机构的软件开发作出可观的贡献。
GunjanDoshi CommunityConnect,Inc，产品开发和流程优化部副总 Marick通过一系列简短实用的例子对Ruby语言进行了解说。
相信Watir用户和其他准备学习Ruby的测试人员一定会乐在其中。
BretPettichord Watir，开发主管 读完本书，你能学会测试自动化，这会让你在你的QA同伴中脱颖而出；你还能学会用Ruby编程，它其乐无穷；你会得到若干有用的实用工具，稍加修改就能满足特定需求。
本书让这一切都来得轻松愉快。
你会成为更有效率的测试人员，并（很可能）非常享受测试的全过程。
GeorgeHawthorne OblomovConsulting咨询师 本书相当出色。
在这样一本相对苗条的书里，既有丰富的内容，又覆盖了很多领域，这让我感觉非常棒。
我有很多超过800页的技术书籍，我一般只能读完前面的一半或2/3，因为它们往往越往后越深奥。
本书却让我一直兴致盎然，我有一份全职工作和一个10个月大的儿子，而我仍在一个星期左右的时间里看完了本书的所有示例代码。
Brian的个性对本书起到了良好作用（比如说“狗窝”那段内容），它使得所述内容更容易理解。
PaddyHealey Aventail，企业系统工程师 本书关于正则表达式的章节、示例和练习就能值回本书的价钱了！
其余的就是你赚到的——那是你不曾想过能够拥有的各种甜点。
不管你是初学乍练还是编程老手，本书都是无价之宝。
那些示例、练习、各种Ruby语言细节、循序渐进的方法和讲解会让你在测试自动化方面进入一个全新的境界。
PaulCarvalho SoftwareTestingandQualityServices，咨询师

编辑推荐

大量精彩的诀窍、指导、警告、例子、隐喻、练习及提问！
对于那些想最大限度地利用测试自动化的软件测试从业者来说，本书极具参考价值。
Grigori Melnik，加拿大卡尔加里大学讲师　　作为测试人员，你用于创建复杂测试数据的时间是否比使用它们的时间还要长?作为业务分析师，你多年读书的结果是否只是让你忙于把数据从报表拷贝到电子表格中?作为一个程序员，你是否每天都不得不巡视版本控制系统的输出来寻找所需文件?
如果你对上述问题的回答有一个为“是”，那么你就是在浪费工作台上的计算机。
应该让计算机做它应该做的事情，而你应该做的事情是思考！
你只需要一门脚本语言（它是免费的！
）和本书（它很便宜！
），并花一些时间通读示例程序并完成练习就足够了。
从本书中，你将学到Ruby的一些基础知识，并了解如何使用测试驱动设计编写稳定可控的脚本。
你还能学到如何使用应用程序框架，它使你编写少量代码即可完成工作。
你还会学到如何处理常见的错误。
你将学会辨明是否会走进死胡同，并学会怎样走出困境。
你甚至能看到那些最常见的键盘输入错误，当真正见到它们时，你会轻松地意识到错在哪里。
■学会让任务自动化。
■获得一些真正有用的脚本程序，它们可以轻松地应用到你的工作中。
■理解编程术语和概念。
■从已有的代码和库中获益。
■跟团队成员更有效地沟通。
Brian Marick拥有英语文学和计算机科学的双重学位，多面手的才能也体现在他的职业生涯中。
有时他是程序员，有时他又是测试人员。
不管在哪个领域，他都比别人做得更好，因为他了解另外一个领域。
在编写《The Craft of Softwa re Testin9》一书时，他就展现了这种才能。
Brian还是《敏捷软件开发》（《Agile Software Development》）的起草者之一，并从此在敏捷方法学（Agile methodologies）方面有深入研究。
因为敏捷方法鼓励跨学科的工作方式，他又一次发现自己不适合固定的领域。
当做咨询工作时，他必须面对各种挑战。
作为测试人员，你用于创建复杂测试数据的时间是否比使用它们的时间还要长？
作为业务分析师，你多年读书的结果是否只是让你忙于把数据从报表拷贝到电子表格中？
作为一个程序员，你是否每天都不得不巡视版本控制系统的输出来寻找所需文件？
如果你对上述问题的回答有一个为“是”，那么你就是在浪费工作台上的计算机。
应该让计算机做它应该做的事情，而你应该做的事情是思考！
你只需要一门脚本语言（它是免费的！
）和本书（它很便宜！
），并花一些时间通读示例程序并完成练习就足够了。
从本书中，你将学到Rubv的一些基础知识，并了解如何使用测试驱动设计编写稳定可控的脚本。
你还能学到如何使用应用程序框架，它使你编写少量代码即可完成工作。
你还会学到如何处理常见的错误。
你将学会辨明是否会走进死胡同，并学会怎样走出困境。
你甚至能看到那些最常见的键盘输入错误，当真正见到它们时，你会轻松地意识到错在哪里。

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