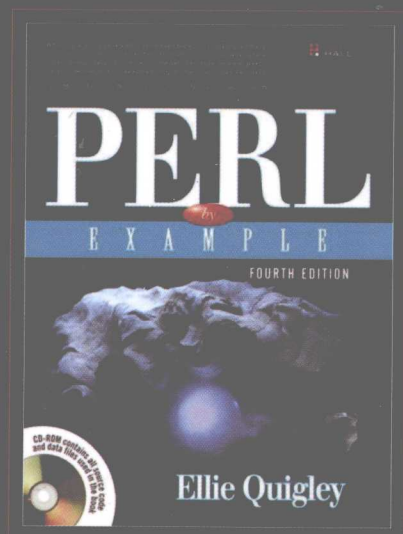


PERL 实例精解

(原书第4版)

PERL by Example, 4E

(美) Ellie Quigley 著
陈宗斌 等译



经典的Perl学习指南
数万Perl程序员的案头必备



机械工业出版社
China Machine Press

华章程序员书库



PERL 实例精解

(原书第4版)

PERL by Example, 4E

(美) Ellie Quigley 著
陈宗斌 等译



机械工业出版社
China Machine Press

本书不仅是一本初学者指导,而且是关于 Perl 的完整指南,它包含了 Perl 应用的很多方面,从正则表达式处理到格式化报表,再到进程间通信等。由于 Perl 原本是为 UNIX 编写的,因此具有一些 UNIX 知识会大大加快学习 Perl 的进程。任何读、写或只是维护 Perl 程序的人都能极大地从本书受益。开发人员可以通过借鉴本书的示例来编写良好的 Perl 程序,甚至可以在实际工作中直接使用本书中的示例程序,从而大大节省开发人员的时间,并且能有效地避免编程错误。

本书内容简洁、示例丰富,非常适合于 Perl 语言初学者阅读,对于 Perl 程序员,也有很大的参考价值。

Simplified Chinese edition copyright ©2008 by Pearson Education Asia Limited and China Machine Press.

Original English language title: Perl by Example (ISBN 978-0-13-238182-6) by Ellie Quigley Copyright ©2008.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc.

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签,无标签者不得销售。

版权所有,侵权必究。

本书法律顾问 北京市展达律师事务所

本书版权登记号: 图字: 01-2008-1796

图书在版编目 (CIP) 数据

PERL 实例精解 (原书第 4 版) / (美) 奎格利 (Quigley, E.) 著; 陈宗斌等译. —北京: 机械工业出版社, 2008.10

(华章程序员书库)

书名原文: Perl by Example, 4E

ISBN 978-7-111-25169-9

I . P... II . ①奎... ②陈... III . PERL 语言—程序设计 IV . TP312

中国版本图书馆 CIP 数据核字 (2008) 第 148588 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 陈佳媛

北京牛山世兴印刷厂印刷 · 新华书店北京发行所发行

2009 年 1 月第 1 版第 1 次印刷

186mm×240mm·45.5 印张

标准书号: ISBN 978-7-111-25169-9

定价: 95.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换
本社购书热线 (010) 68326294

对 Ellie Quigley 著作的赞誉

“我整个周末都捧着一本《JavaScript by Example》。我想感谢作者撰写了这本好书，它让 JavaScript 变得易于理解了。作为拥有多年经验的开发人员，我经常感到 JS 就像‘床铺下的怪物’一样不可琢磨。作者撰写的这本书则解答了很多有关 JS 内部实现的问题，这些问题已经悬在我心上很长时间，却一直难以启齿求教他人。现在，我所需要的不过是一本涵盖 Ajax 与 Coldfusion 的书而已。再次感谢作者提供了一本出色的书。”

——Chris Gomez, Web 服务经理, Zunch Worldwide, Inc.

“我曾经阅读过您的《UNIX Shells by Example》。必须承认的是，它真是棒极了。其他大多数书都没有涵盖全部的 Shell 类型，当读者置身于某个只使用 tcsh、bash 和 korn 的机构中时，就会发现一切还是那么困难。如果读者想要学习各种 shell 并洞悉其细微差别的话，这本书就是不可或缺的……我发这封电子邮件的目的只是想让您知道：您所做的工作有多么伟大！”

——Farogh-Ahmed Usmani, B.Sc. (Honors), M.Sc., DIC,
项目顾问 (Billing Solutions), Comverse

“我学习 Perl 刚刚有两个月；因此只有很少的 shell 脚本编程经验。一开始，我尝试着学习 O'Reilly 出版的《Learning Perl》。这是一本好书，但它的示例太少了。然后我转而研读 Larry Wall 撰写的《Programming Perl》。这本书对于中高级水平的读者而言非常有用，但对我这种 Perl 初学者则无济于事。最后我拿起了《Perl by Example, Third Edition》。这本书是一本超好的编程教材。我曾经阅读过大量计算机书籍，觉得这本书可以归入头两名之列。书中提供的实例都非常精彩。在每个实例中，作者展示了整段代码及其各行的输出内容，并对每一行都进行了解释。”

——Dan Patterson, 软件工程师, GuideWorks, LLC

“Ellie Quigley 编写了一本出色的 Perl 入门教材，我就是使用这本教材从头开始学习 Perl 语言的。读者所要做的只是学习和体验所有的实例，不久之后就会变得熟悉这门语言了。尽管我在学生时代使用的是 Wall 等人合著的《Programming Perl》，但还是觉得 Quigley 的著作是最为有益的参考书。”

——Casey Machula, 系统支持分析师, 北亚利桑那大学健康及人类服务学院

“本人的书架上可以找到 11 本 Perl 编程教材。《Perl by Example, Third Edition》没有出现在书

架上，而是驻扎在我的办公桌上，因为我每天都需要用到它。在购买这本书时，我并没有太多编程经验，而且过去使用的还是 COBOL。因此，我是如假包换的 Perl 初学者。那个时候，我购买了很多本有关 Perl 的书籍，却发现其中没有一本满足我的需求。虽然不是什么专家，但我已经在这本书上折出了好多折角，其中每个折角都代表了我学会并牢记的一课内容。

我认为，本书是市面上最好的 Perl 教材，它适合我们每一个人，从初学者到每天都在使用 Perl 的程序员，莫不如此。”

——Bill Maples, 网络设计工具和自动化分析师，
福德信息技术服务有限公司

“我们正在重新撰写 OS 脚本课程的介绍部分，并选用您的这本书作为该课程的教材。这本书写得非常出色。数年之前我们也有过同样的感受（那时还是第 2 版）。书末尾处的有关调试和系统管理的章节让我们受益匪浅。”

——Jim Leone 哲学博士，信息技术系教授罗切斯特技术学院

“Quigley 的著作揭示了 PHP 的主要用途，即编写某种前端用户界面程序，并连接到后端 MySQL 数据库上。这两项技术都是免费和开源的，二者结合起来自然就日益流行了。特别是当前端需要利用内嵌的 PHP 命令生成 HTML Web 页面时更是如此。

并非每个实例都同时涉及 PHP 与 MySQL，不过所有的实例都与 PHP 相关。其中很多实例都展示了如何在 HTML 文件中使用 PHP。譬如编写用户自定义函数、嵌套函数，或者创建或使用各种函数库。函数是 PHP 的核心概念之一，它能把用户从各种元素标记中解脱出来。函数还能让用户方便地使用其他 PHP 编程人员提供的代码。如果用户是某个编程小组中的一员，那这一点就显得尤为重要，因为通过它就能以某种方式分配整个编码任务。”

——Wes Boudville 博士，CTO, Metaswarm Inc.

译者序

Perl 程序设计语言最初是 UNIX 系统管理员的工具,用于处理日常的繁琐事务。时至今日,Perl 已发展成一套功能齐备的程序语言,横跨所有的操作平台,其用途已不仅仅限于处理日常事务,目前已经扩展到涵盖了 Web 程序设计、数据库操作、XML 处理以及系统管理等方面。

顾名思义,本书使用示例来全面介绍 Perl 语言,其内容包含 Perl 的所有主题。本书对每个主题都使用了示例说明,从 Perl 变量、正则表达式,到编写报表、CGI 脚本和网络应用,并对示例程序进行了解释。读者可以很容易地通过这些示例掌握 Perl 语言的语法。附录含有完整的函数和定义、命令行开关、特殊变量、常见模块和 Perl 调试器的列表,全功能的、面向对象的 CGI 程序,一些有用的脚本以及很有帮助的 HTML 教程。

本书不仅是一本初学者指导,而且是关于 Perl 的完整指南,它包含了 Perl 应用的很多方面,从正则表达式处理到格式化报表,再到进程间通信等。由于 Perl 原本是为 UNIX 编写的,具有一些 UNIX 知识会大大加快学习 Perl 的进程。任何读、写或只是维护 Perl 程序的人都能极大地从本书受益。开发人员可以通过借鉴本书的示例来编写良好的 Perl 程序,甚至可以在实际工作中直接使用本书中的示例程序,从而大大节省了开发人员的时间,并且能有效地避免编程错误。

本书内容简洁、示例丰富,非常适合于 Perl 语言初学者阅读,对于 Perl 程序员,也有很大的参考价值。

参加本书翻译的人员有:陈宗斌、乔健、张景友、易小丽、陈婷、管学岗、王新彦、金惠敏、张海峰、徐晔、戴锋、张德福、张士华、张锁玲、杜明宗、高玉琢、王涛、申川、孙玲、李振国、高德杰、宫飞、侯经国、刘淑妮、张春林、李大成、程明、张路红、张淑芝、孙先国、刘冀得、梁永翔、张广东、郁琪琳、邵长凯、蒲书箴、潘曙光、刘瑞东、李军、焦敬俭、赵中元、金鑫、赵宏伟、张宏顺、尹周、王开年、贾震、陆晓萍、金国良、俞群。

由于时间紧迫,加之译者水平有限,错误在所难免,恳请广大读者批评指正。

本书在翻译过程中,参考了 Perl 语言官方网站 www.perl.com 上的相关资料,以及 Perl 语言社区的一些书籍和文章。同时,本书还参考了 Perl 语言社区的一些书籍和文章。同时,本书还参考了 Perl 语言社区的一些书籍和文章。

本书在翻译过程中,参考了 Perl 语言官方网站 www.perl.com 上的相关资料,以及 Perl 语言社区的一些书籍和文章。同时,本书还参考了 Perl 语言社区的一些书籍和文章。

前言

读者或许想知道，为什么本书又出新版本？Perl 5 的变化并不太大；事实上，自从本书第 3 版之后，其变化就已经很小了。Perl 6 还没有正式发布，那作者为什么不再等等呢？假设读者在 6 年前购买了一台 Whirlpool 洗衣机。这台洗衣机一直运行得很好。但从那时起，洗涤的衣物则是换了一茬又一茬。读者现在可能已是时尚一族，拥有出自名家之手的被褥或拖鞋，并用上了低刺激性、无毒无害且能生物降解的洗衣粉。而这些都是当初在购买这台洗衣机时无法预料到的。尽管 Perl 5 的变化很小，但整个计算机世界却已经翻天覆地。新革新、新技术、新应用乃至新时尚不断涌现，程序则不得不迎合这些变化。当从 GenBank 序列数据库中分析数据、为 iPhone 编写程序、在“myspace”上撰写个人 blog、或者适应新的 Windows Vista 时，都会涉及某些计算机程序，它们很可能就是由 Perl 编写的。无论如何，我们都必须跟上时代的步伐。这个新版本恰恰是为了达到上面这个目标。

笔者目前正在位于加州 Sunnyvale 的 UCSC^①分校讲授 Perl，台下的学员则是来自硅谷各地的专业人士。笔者总会在第一节课上提出这样一个问题：“你为什么要学习 Perl？”回答是各式各样的，譬如：“我们公司在 Web 上有个拍卖站点，而我是网络管理员，需要用 Perl 和 Apache 处理订单信息，并发送给 Oracle”，“我在 Stanford 的基因研究小组工作，需要处理大量数据。我听说只要学好 Perl，就不用再找其他程序员来干这个了”，“我是 UNIX/Linux 管理员，而我们公司刚刚决定所有管理脚本都必须更改为 Perl 形式”，或者“我刚刚遭到解雇，听说在个人简历里加上 Perl 是绝对必要的”。最让笔者惊讶的是，这些 Perl 学员来自于各种不同的行业：工程师、科学家、基因研究者、气象学家、经理、销售员、程序员、教师、硬件人员、学生、证券商、各类管理员、作家、银行家和艺术家等。Perl 能为所有人服务，并能胜任各种工作。

不论是谁都得承认：一图胜千言，实例也是如此。本书致力于通过完整而简洁的实例片断来向读者灌输有关 Perl 的知识。这些脚本实例的每一行都拥有自己的行号，其中一些重要的行还通过黑体字突出显示出来。后面是带有行号的程序输出内容，其行号对应于脚本的行号。再后则是单独的对每一行输出的解释。这些实例都短小精悍，并紧扣正在介绍的主题。由于本书主要是用作 Perl 课堂上的学生指南，因此书中各个主题都进行了合理的划分。每一章都建立在前一章的基础上，并尽量避免引用后面的内容，同时在主题和主题之间维护较好的逻辑次序。各章结尾处都提供了一些练习。读者可以在华章网站上找到所有的书中实例，网址为 www.hzbook.com。这些实例都已经在主流平台上得到了充分的测试。

本书并不仅仅是一本初学者指南，而且是一本对 Perl 的完整指南。它涵盖了 Perl 的方方面面，从正则表达式的处理，到报表的格式化，乃至进程间通信。它将教会读者使用 Perl，并在此过程中

① University Of California, Santa Cruz.

介绍大量有关 Linux 和 Windows 的知识。由于 Perl 最初是针对 UNIX 系统编写的，因此拥有一些 UNIX 背景知识将大大加速学习的进程，不过这并不是说要读者成为 UNIX 专家。任何期望读、写或仅仅是维护 Perl 程序的人士都能从本书中受益匪浅。

Perl 拥有丰富的函数，能够处理字符串、数组、系统接口、网络等。为方便读者理解这些函数的功能，本书在展示函数示例程序前还提供了一些有关该函数的背景信息。这样就能避免出现读者为了获取背景知识、参数含义和函数用途而不得不频繁查阅手册和其他书籍的尴尬情况。

本书附录中含有有关函数定义、命令行开关、特殊变量、常见模块和 Perl 调试器的完整列表，给出了一段介绍 BioPerl 模块的说明性文字，还提供了一组有关 `mod_perl` 的说明文档，其中 `mod_perl` 是一种快速建立服务器端 Perl 脚本的方式，它能满足通用网关接口（Common Gateway Interface, CGI）的实际需求。

笔者在过去的 30 年间一直在教书，因此非常了解人们是如何学习的。在讲授了超过 14 年的 Perl 之后，笔者发现很多 Perl 初学者往往会在自学编程时感到灰心丧气。笔者发现，对于绝大多数人而言，最好的学习方法莫过于从简洁的小示例和实践中入门。因此，笔者撰写了这样一本书，以帮助自己和学生来学习 Perl，同样也帮助读者学习它。随着 Perl 的不断发展，本书也在不断修订。在这本最新出版的第 4 版中，新增了一章介绍基于 MySQL 的 Perl 与 DBI 组件的内容，修订了 Perl 对象相关章节，并在其他章节中添加了很多示例和解释，以便让它们显得更为新颖和有趣。本书附录部分也经过了修订，包含了有关 BioPerl 和 `mod_perl` 的内容。在本书中，读者将不仅学会 Perl，还将节省大量时间。至少笔者的很多学生和读者都这么说。究竟本书怎么样，读者您说了算。

致 谢

笔者希望对下列各位同仁为本书第 4 版的贡献予以致谢。

感谢 Dmitri Korzh 和 Techne Group, 他们通过 FrameMaker 对笔者粗糙的草稿进行了编辑、格式化和索引, 并把它变成了一本看上去非常专业的图书。

笔者还要感谢 Oleg Orel, 一位来自 NetApp 的杰出学生。他为有关对象的章节编写了说明“闭包”概念的示例程序, 并帮助笔者解决了从 CPAN 下载模块时碰到的问题。

感谢 Mark Taub, 一位倍受赞誉的主编。从签订合同到全书出版, 他在其中每个步骤都表现得那么酷。Mark 总能让看上去不可能的工作成为可能; 他能心平气和地讨论不可能达成的最终期限, 能时刻保持一贯的压力, 还能在错过最终期限时从不歇斯底里, 而是默默无闻地完成他的目标, 并总是拥有些许微妙的幽默感。感谢你, Mark, 你是本书新版本背后的驱动力。

当然, 如果没有了 Perl 先驱们 (Larry Wall、Randal Schwartz 以及 Tom Christiansen) 的贡献, 本书介绍的一切内容都将无从谈起。他们的著作是学习 Perl 的必读读物, 包括 Randal Schwartz 撰写的《*Learning Perl*》以及由 Larry Wall、Tom Christiansen 和 Jon Orwant 共同编写的《*Programming Perl*》。

最后, 非常感谢本书在世界各地的学生, 他们为本书找出了大量瑕疵, 并让各个主题讨论总是那么热火朝天。

目 录

对 Ellie Quigley 著作的赞誉	1.0
译者序	5.0
前 言	8.0
致 谢	1.0.0
第 1 章 实用摘要和报表语言	1
1.1 什么是 Perl	1
1.2 什么是解释语言	2
1.3 Perl 的用户	2
1.3.1 Perl 的版本	3
1.3.2 什么是 Perl 6	3
1.4 如何获得 Perl	4
1.5 什么是 CPAN	7
1.6 Perl 文档	8
1.6.1 Perl 的 man 页面	8
1.6.2 HTML 文档	9
1.7 读者应当学到的知识	9
1.8 下章简介	10
第 2 章 Perl 快速入门	11
2.1 快速入门和速查手册	11
2.1.1 给程序员的提示	11
2.1.2 给非程序员的提示	11
2.1.3 Perl 语法和结构	11
2.2 本章小结	23
2.3 下章简介	23
第 3 章 Perl 脚本	24
3.1 创建脚本	24
3.2 脚本	25
3.2.1 启动	25
3.2.2 选择文本编辑器	26

3.2.3 为 Perl 脚本取名	26
3.2.4 语句、空白和换行	27
3.2.5 注释	27
3.2.6 Perl 语句	28
3.2.7 使用 Perl 内建函数	28
3.2.8 执行脚本	28
3.2.9 脚本实例	29
3.2.10 可能出现的错误	30
3.3 从命令行使用 Perl	31
3.3.1 -e 开关	31
3.3.2 -n 开关	31
3.3.3 -c 开关	33
3.4 读者应当学到的知识	33
3.5 下章简介	34
第 4 章 获得打印句柄	35
4.1 文件句柄	35
4.2 字 (Word)	35
4.3 print 函数	35
4.3.1 引号	36
4.3.2 实量 (常量)	38
4.3.3 打印实量	40
4.3.4 warning 编译指示符和	42
-w 开关	42
4.3.5 diagnostics 编译指示符	44
4.3.6 strict 编译指示符	44
4.4 printf 函数	45
4.4.1 sprintf 函数	48
4.4.2 无引号打印: here 文档	48
4.5 读者应当学到的知识	51
4.6 下章简介	51

第 5 章 变量	53	5.5.4 对散列进行排序	93
5.1 Perl 变量简介	53	5.5.5 delete 函数	98
5.1.1 类型	53	5.5.6 exists 函数	99
5.1.2 作用域和包	53	5.6 有关散列的更多内容	99
5.1.3 命名规范	54	5.6.1 从文件载入散列	99
5.1.4 赋值语句	54	5.6.2 特殊散列变量	100
5.1.5 引号规范	55	5.6.3 上下文 (Context)	102
5.2 标量、数组和散列	59	5.7 读者应当学到的知识	103
5.2.1 标量型变量	59	5.8 下章简介	104
5.2.2 数组	61	第 6 章 运算符	106
5.2.3 散列	67	6.1 关于 Perl 运算符	106
5.2.4 复杂数据结构	70	6.2 混合数据类型	106
5.3 从 STDIN 读取输入	72	6.3 优先级和结合性	108
5.3.1 把输入内容赋值给标量型变量	72	6.3.1 赋值运算符	109
5.3.2 chop 与 chomp 函数	73	6.3.2 关系运算符	111
5.3.3 read 函数	74	6.3.3 相等性运算符	113
5.3.4 getc 函数	75	6.3.4 逻辑运算符 (短路运算符)	115
5.3.5 将输入内容赋予数组	75	6.3.5 逻辑与运算符	117
5.3.6 将输入内容赋予散列	76	6.3.6 算术运算符	118
5.4 数组函数	77	6.3.7 自动递增与自动递减运算符	119
5.4.1 chop 和 chomp 函数	77	6.3.8 位逻辑运算符	121
(用于列表)	77	6.3.9 条件运算符	123
5.4.2 exists 函数	78	6.3.10 范围运算符	125
5.4.3 delete 函数	78	6.3.11 特殊字符串运算符和函数	125
5.4.4 grep 函数	78	6.3.12 算术函数	127
5.4.5 join 函数	79	6.4 读者应当学到的知识	130
5.4.6 map 函数	80	6.5 下章简介	131
5.4.7 pack 和 unpack 函数	81	第 7 章 条件	133
5.4.8 pop 函数	82	7.1 控制结构、块与复合语句	133
5.4.9 push 函数	83	7.2 循环	138
5.4.10 shift 函数	83	7.2.1 while 循环	138
5.4.11 splice 函数	84	7.2.2 until 循环	139
5.4.12 split 函数	85	7.2.3 for 循环	141
5.4.13 sort 函数	88	7.2.4 foreach 循环	143
5.4.14 reverse 函数	90	7.2.5 循环控制	146
5.4.15 unshift 函数	91	7.2.6 switch 语句	152
5.5 散列 (关联数组) 函数	91	7.3 读者应当学到的知识	155
5.5.1 keys 函数	91	7.4 下章简介	155
5.5.2 values 函数	92	第 8 章 正则表达式——模式匹配	156
5.5.3 each 函数	93	8.1 什么是正则表达式	156

8.2 表达式修饰符与简单语句	156	第 11 章 子例程与函数	250
8.2.1 条件修饰符	157	11.1 子例程 / 函数	250
8.2.2 DATA 文件句柄	158	11.2 参数传递	254
8.2.3 循环修饰符	160	11.2.1 原型	260
8.3 正则表达式运算符	161	11.2.2 返回值	262
8.3.1 m 运算符与匹配	162	11.2.3 上下文和子例程	263
8.3.2 s 运算符与替换	167	11.3 按引用调用	265
8.3.3 模式绑定运算符	174	11.3.1 符号引用 — typeglob	265
8.4 读者应当学到的知识	179	11.3.2 硬引用 — 指针	269
8.5 下章简介	179	11.3.3 自动加载	273
第 9 章 正则表达式元字符	181	11.3.4 BEGIN 和 END 子例程 (开始与结束)	275
9.1 正则表达式元字符	181	11.3.5 subs 函数	275
9.1.1 表示单个字符的元字符	183	11.4 读者应当学到的知识	276
9.1.2 空白元字符	190	11.5 下章简介	276
9.1.3 重复模式匹配元字符	192	第 12 章 模块化、打包并发送到库	278
9.1.4 tr 或 y 函数	213	12.1 包和模块	278
9.2 Unicode	216	12.1.1 引言	278
9.3 读者应当学到的知识	218	12.1.2 一个类比	278
9.4 下章简介	218	12.1.3 定义	278
第 10 章 获得文件句柄	220	12.1.4 符号表	279
10.1 用户定义文件句柄	220	12.2 标准 Perl 库	284
10.1.1 打开文件: open 函数	220	12.2.1 @INC 数组	284
10.1.2 打开文件读取	220	12.2.2 包和 .pl 文件	286
10.1.3 打开文件写入	224	12.2.3 模块和 .pm 文件	290
10.1.4 Win32 二进制文件	225	12.2.4 导出和导入	290
10.1.5 打开文件追加	226	12.2.5 如何 “use” 来自标准 Perl 库的 模块	295
10.1.6 select 函数	227	12.2.6 使用 Perl 创建自己的模块	298
10.1.7 使用 flock 为文件加锁	228	12.3 来自 CPAN 的模块	299
10.1.8 seek 和 tell 函数	229	12.4 读者应当学到的知识	305
10.1.9 打开文件读写	233	12.5 下章简介	306
10.1.10 打开管道	234	第 13 章 这项工作是否需要引用吗	308
10.2 参数传递	239	13.1 什么是引用, 什么是指针	308
10.2.1 ARGV 数组	239	13.1.1 符号引用和硬引用	308
10.2.2 ARGV 与 Null 文件句柄	241	13.1.2 硬引用, 指针	310
10.2.3 eof 函数	243	13.1.3 引用和匿名变量	312
10.2.4 -i 开关: 原位编辑文件	245	13.1.4 嵌套数据结构	313
10.3 文件测试	246	13.1.5 引用和子例程	318
10.4 读者应当学到的知识	248	13.1.6 文件句柄引用	320
10.5 下章简介	249		

13.1.7 ref 函数	321	14.8 下章简介	373
13.2 读者应当学到的知识	323	第 15 章 神奇的 Tie 和 DBS	377
13.3 下章简介	323	15.1 连接变量与类	377
第 14 章 面向对象的 Perl	325	15.1.1 tie 函数	377
14.1 OOP 范例	325	15.1.2 预定义方法	378
14.1.1 回顾包与模块	325	15.1.3 连接标量	378
14.1.2 一些面向对象的专用 术语	326	15.1.4 连接数组	380
14.2 类、对象和方法	326	15.1.5 连接散列	382
14.2.1 现实世界	326	15.2 DBM 文件	386
14.2.2 步骤	327	15.2.1 创建并赋值 DBM 文件	387
14.2.3 类和私有性	327	15.2.2 从 DBM 文件中检索数据	389
14.2.4 对象	328	15.2.3 从 DBM 文件删除项	390
14.2.5 bless 函数	330	15.3 读者应当学到的知识	392
14.2.6 方法	331	15.4 下章简介	392
14.2.7 面向对象的模块样式	332	第 16 章 CGI 和 Perl: 超级活力 双雄	393
14.2.8 多态性和动态绑定	341	16.1 静态和动态 Web 页面	393
14.2.9 析构函数和垃圾收集	346	16.2 工作原理	395
14.3 匿名子例程、闭包和私有性	347	16.3 使用 HTML 创建 Web 页面	399
14.3.1 什么是闭包	348	16.4 HTML 和 CGI 如何协作	403
14.3.2 闭包和对象	349	16.4.1 简单 CGI 脚本	403
14.4 继承	353	16.4.2 错误日志文件	405
14.4.1 @ISA 数组和调用方法	353	16.5 控制信息出入 CGI 脚本	406
14.4.2 \$AUTOLOAD、sub AUTOLOAD 和 UNIVERSAL	355	16.6 CGI 和表单	409
14.4.3 派生类	357	16.6.1 表单输入类型	410
14.4.4 多重继承	362	16.6.2 创建 HTML 表单	410
14.4.5 重写父类方法	362	16.6.3 GET 方法	413
14.5 公共用户接口: 文档类	364	16.6.4 处理编码数据	416
14.5.1 pod 文件	364	16.6.5 总结	418
14.5.2 pod 命令	365	16.6.6 POST 方法	421
14.5.3 如何使用 pod 解释器	366	16.6.7 处理电子邮件	424
14.5.4 将 pod 文档转译为文本	367	16.7 CGI.pm 模块	427
14.5.5 将 pod 文档转译为 HTML	368	16.7.1 简介	427
14.6 使用 Perl 库中的对象	368	16.7.2 优点	428
14.6.1 另眼看标准 Perl 库	368	16.7.3 使用 CGI.pm 编程的两种形式	428
14.6.2 一个来自标准 Perl 库的面向 对象模块	369	16.7.4 重要警示	430
14.6.3 使用标准 Perl 库中的模块	371	16.7.5 HTML 表单方法	431
14.7 读者应当学到的内容	372	16.7.6 CGI.pm 如何处理表单	436
		16.7.7 CGI.pm 表单元素	441
		16.7.8 生成表单输入字段的方法	442

16.7.9 错误处理	453	18.1.6 更改目录——chdir 函数	532
16.7.10 HTTP 首部方法	455	18.1.7 通过目录文件句柄访问目录	533
第 17 章 当 Perl 遇见 MySQL:		18.1.8 权限和所有权	535
完美的连接	460	18.1.9 硬链接和软链接	538
17.1 简介	460	18.1.10 更改文件名	541
17.2 什么是关系型数据库	460	18.1.11 更改访问与修改时间	541
17.2.1 客户端/服务器型数据库	461	18.1.12 文件统计量	542
17.2.2 关系型数据库的组成部分	462	18.1.13 底层 I/O 文件	544
17.2.3 通过 SQL (结构化查询语言)		18.1.14 压缩和解压缩数据	546
访问数据库	466	18.2 进程	550
17.3 MySQL 入门	468	18.2.1 UNIX 进程	551
17.3.1 为何选用 MySQL	468	18.2.2 Win32 进程	552
17.3.2 安装 MySQL	468	18.2.3 环境 (UNIX 和 Windows)	552
17.3.3 连接 MySQL	468	18.2.4 进程和文件句柄	554
17.3.4 图形化用户界面	470	18.2.5 进程优先级和 Nice 值	556
17.3.5 寻找数据库	472	18.2.6 口令信息	557
17.3.6 基本命令入门	474	18.2.7 时间和进程	561
17.4 什么是 Perl DBI	487	18.2.8 UNIX 进程创建	564
17.4.1 安装 DBI	488	18.2.9 Win32 进程创建	568
17.4.2 DBI 类方法	493	18.3 与操作系统交互的其他途径	570
17.4.3 如何使用 DBI	495	18.3.1 syscall 函数和 h2ph 脚本	571
17.4.4 连接和断开数据库	495	18.3.2 命令替换——反引号	572
17.4.5 disconnect() 方法	497	18.3.3 Shell.pm 模块	572
17.4.6 准备语句句柄并获取结果	497	18.3.4 system 函数	573
17.4.7 处理引号	500	18.3.5 here 文档	575
17.4.8 获取错误消息	501	18.3.6 Globbing (文件名扩展和通配符)	575
17.5 不返回数据的语句	509	18.4 错误处理	577
17.6 事务	513	18.4.1 die 函数	577
17.7 使用 CGI 和 DBI 选择并显示记录项	515	18.4.2 warn 函数	578
17.8 还有哪些内容	518	18.4.3 eval 函数	578
17.9 读者应当学到的知识	519	18.5 信号	581
17.10 下章简介	520	18.6 读者应当学到的知识	584
第 18 章 与系统交互	523	18.7 下章简介	585
18.1 系统调用	523	第 19 章 借助形象行编写报表	586
18.1.1 目录和文件	524	19.1 模板	586
18.1.2 目录和文件属性	526	19.1.1 定义模板的步骤	586
18.1.3 查找目录和文件	528	19.1.2 更改文件句柄	588
18.1.4 创建目录——mkdir 函数	530	19.1.3 页眉格式	589
18.1.5 删除目录——rmdir 函数	531	19.1.4 select 函数	593

19.1.5 多行字段	595	20.5.2 套接字域	607
19.1.6 填充字段	595	20.5.3 创建套接字	608
19.1.7 动态报表编写	597	20.5.4 绑定地址和套接字名	609
19.2 读者应当学到的知识	598	20.5.5 创建套接字队列	609
19.3 下章简介	599	20.5.6 等待客户端请求	609
第 20 章 通过网络发送	600	20.5.7 建立套接字连接	610
20.1 网络与 Perl	600	20.5.8 关闭套接字	610
20.2 客户端/服务器模型	600	20.6 客户端/服务器程序	611
20.3 网络协议 (TCP/IP)	600	20.6.1 同一台计算机上面面向连接的 套接字	611
20.3.1 以太网协议 (硬件)	600	20.6.2 远程计算机 (Internet 客户端和 服务器) 上面面向连接的 套接字	614
20.3.2 Internet 协议 (IP)	601	20.7 Socket.pm 模块	617
20.3.3 传输控制协议 (TCP)	601	20.8 读者应当学到的知识	621
20.3.4 用户数据报协议 (UDP)	601	附录 A Perl 内建函数、编译指示、 模块和调试器	622
20.4 网络寻址	601	附录 B SQL 语言入门	654
20.4.1 以太网地址	601	附录 C Perl 和生物学	692
20.4.2 IP 地址	602	附录 D 强悍与速度: CGI 和 mod_Perl	699
20.4.3 端口号	602		
20.4.4 Perl 协议函数	602		
20.4.5 Perl 的服务器函数	603		
20.4.6 Perl 的主机信息函数	605		
20.5 套接字	607		
20.5.1 套接字的类型	607		

第1章 实用摘要和报表语言

1.1 什么是 Perl

“懒惰、性急和傲慢。伟大的 Perl 程序员拥有这些优点。”

——Larry Wall

Perl 是一种多用途的开源（免费软件）解释型语言，由称为 Perl Porters 的核心开发团队维护和改进。它主要用作脚本语言，并且运行在众多平台上。尽管 Perl 最初是为 UNIX 操作系统设计的，但是它以其可移植性以及现在与大多数操作系统捆绑在一起而著名。这些操作系统包括 RedHat Linux、Solaris、FreeBSD、Macintosh 等。由于 Perl 的通用性，它通常称为程序设计语言的“瑞士军刀”。

Larry Wall 编写 Perl 语言来管理散布在网络中的日志文件和报表。依据 Wikipedia.org 上面的说法：“Perl 最初命名为 ‘Pearl’，它出自于 Gospel of Matthew（《圣经马太福音》）中的 ‘Parable of the Pearl’（珍珠的寓言）。”该寓言的简要内容如下：一位商人寻找珍珠，他找到一颗如此贵重、漂亮的珍珠，以至于他情愿倾其所有来购买它。最后，他甚至比以往更富有。无论你怎么解释这则寓言，它都具有非常积极的寓意。

但是在 1987 年推出其官方发布版时，删去了“Pearl”中的字母“a”，自此将该语言称为“Perl”，后来又把它称为实用摘要和报表语言（Practical Extraction and Report Language），还有一些人把它称为病态折衷垃圾列表器（Pathologically Eclectic Rubbish Lister）。你很快将会看到，Perl 确实远远胜过实用报表语言或折衷垃圾列表器。Perl 使编程简单、灵活和快速，因此使用它的人都会喜欢它。其用户从经验丰富的程序员一直到只具有很少计算机知识的初学者，而且用户的数量也在飞速增长。^①

Perl 传承自 UNIX。Perl 脚本在功能上类似于 UNIX awk、sed、shell 脚本和 C 程序。shell 脚本主要由 UNIX 命令构成，Perl 脚本则不然。sed 和 awk 用于编辑和报告文件，但是 Perl 无需执行文件即可工作。C 没有 shell、sed 和 awk 的任何模式匹配和通配元字符，而 Perl 却有扩展字符集。Perl 最初用于操作文件中的文本、从文件中提取数据和编写报表，但经过不断的发展，它现在可以操作进程、执行网络任务、处理 Web 页面、与数据库通信，以及分析科学数据。Perl 确实是程序设计语言的“瑞士军刀”，任何人都可以使用它。

① 当指语言时，将 Perl 拼写为“Perl”；当指解释器时，则拼写为“Perl”。

本书中的示例是在 Solaris、Linux、Macintosh UNIX 和 Win32 系统上创建的。

Perl 通常与 O'Reilly Media 的商标即骆驼标志相关联, O'Reilly Media 出版了第一本关于 Perl 的书籍, 书名是《Programming Perl》, 作者是 Larry Wall 和 Randal Schwartz, 该书称为“Camel Book”(骆驼书)。

1.2 什么是解释语言

在编写 Perl 程序时, 读者需要准备两样工具: 一个文本编辑器和一个 Perl 解释器。读者可以从许多 Web 站点上下载到后者, 譬如 Perl.org、cpan.org 以及 activestate.com。和 C++、Java 等编译语言不同, 读者无需在执行程序之前将其编译成机器能理解的代码。Perl 解释器会代劳这一切, 它能完成程序的编译、解释和执行工作。像 Perl 这样的解释语言有很多优点, 首先, 它能运行在几乎任何平台上; 其次, 它相对易于学习; 最后, 它的速度很快, 并且灵活度很高。

像 Python、Java 和 Perl 等解释语言都用到了中间代码, 以便将编译和解释这两个过程联系起来。它首先会把用户提供的代码编译成一种内部压缩格式, 称之为字节码(bytecode)或连接代码(threaded code), 然后交给解释器执行。在运行 Perl 程序时, 读者会观察到两个阶段: 编译阶段和执行阶段, 后一阶段才会生成程序结果。如果程序中存在语法错误, 譬如关键字拼写错误或缺了引号, 编译器就会报错。即使通过了编译, 程序在开始执行时也可能出现其他问题。成功通过上述两个阶段后, 才可以做其他事情, 譬如改进程序或者提升程序性能等。

此外, 解释器还提供了许多命令行开关项(选项)以控制其行为模式, 譬如语法检查、发送警告信息、遍历文件、执行语句、打开调试器等。本书后面章节将详细介绍这些开关项。

1.3 Perl 的用户

由于 Perl 内建函数能方便地处理进程和文件, 再加上它是可移植的(即能运行在多种不同的平台上), 因此 Perl 深受那些需要同时监控多种不同系统的系统管理员的欢迎。万维网(World Wide Web)的迅速发展大大提高了人们对 Perl 的兴趣。它现在已经成为最流行的 CGI 脚本编写语言, 负责动态生成 Web 页面。即使现在出现了像 ASP.NET 之类的其他专用 Web 页面处理语言, Perl 还是一如既往地受到系统管理员、数据库管理员、科学家、遗传学家以及所有需要从文件里收集和处理数据的用户的欢迎。

任何人都可以使用 Perl。不过倘若读者已经熟悉 UNIX Shell 脚本、C 语言或其他从 C 派生的语言(例如 C++ 和 Java)的话, 学起 Perl 来会更容易一些。对上述用户而言, 转到 Perl 语言的难度相对较小。而对于那些缺乏编程经验的读者, 则需要多花些时间来学习。不过只要学会了 Perl, 您可能就没有必要去使用其他语言了。

如果读者熟悉像 awk、grep、sed 和 tr 这样的 UNIX 工具, 就知道它们的语法其实各不相同, 其选项和参数处理方式也不同, 甚至不同工具所遵循的规则都互不相同。如果读者是一位 Shell 程序员, 往往需要经历一连串艰苦的学习过程, 学会使用各种应用工具、shell 元字符、正则表达式元字