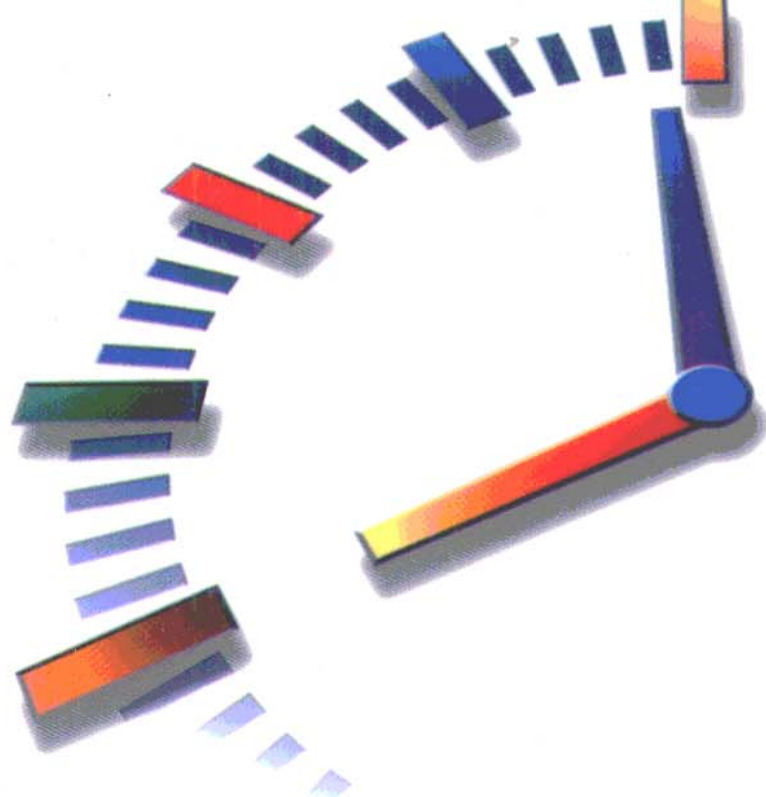


*Sams Teach
Yourself Perl
in 24 Hours*



自学通系列

(美) Clinton Pierce 著
王建华 等译

Perl 编程

24 学时教程

SAMS

机械工业出版社
China Machine Press

TP312PE145

P49

自学通系列

Perl 编程24学时教程

(美) Clinton Pierce 著

王建华 等译

本书附盘可从本馆主页 <http://lib.szu.edu.cn/>
上由“馆藏检索”该书详细信息后下载,
也可到视听部复制



机械工业出版社
China Machine Press

本书循序渐进、深入浅出地介绍了Perl的主要特性及用Perl进行CGI编程的知识。全书包括24个学时内容和一个附录，共分四部分。第一部分主要讲述Perl的基本概念，第二部分重点介绍Perl的一些高级特性，第三部分介绍如何使用Perl进行CGI编程，第四部分(即附录)讲述如何在不同的操作系统下安装Perl的各个模块。

本书结构清晰，讲解透彻，通俗易懂，对于Perl用户来说是一本非常实用的入门参考书。

Clinton Pierce: Sams Teach Yourself Perl in 24 Hours.

Authorized translation from the English language edition published by Sams Publishing, an imprint of Macmillan Computer Publishing U. S. A.

Copyright © 2000 by Sams Publishing.

All rights reserved.

Chinese simplified language edition published by China Machine Press.

Copyright © 2000 by China Machine Press.

本书中文简体字版由美国麦克米兰公司授权机械工业出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2000-0079

图书在版编目(CIP)数据

Perl 编程24学时教程/(美)皮尔斯(Pierce, C.)著;王建华等译.-北京:机械工业出版社,2000.5

(自学通系列)

ISBN 7-111-07987-6

I. P… II. ①皮…②王… III. Perl 语言-程序设计-教材 IV. TP312

中国版本图书馆CIP数据核字(2000)第19567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:李新阳

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2000年5月第1版第1次印刷

787mm×1092mm 1/16·19印张

印数:0 001-6 000册

定价:45.00元(附光盘)

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

前言

Arthur C. Clarke曾说：“任何尽善尽美的技术实际上魔力无穷。”

有一句话请铭记心中，那就是计算机编程并无奥妙可言。

与看上去非常神奇的任何事情一样，它总是有一种诀窍可以使用的，计算机编程也不例外。你需要的是某些分析思考的技能，对知识的渴求，以及需要一定的时间来学习Perl。与从事任何工作一样，学习计算机编程的最好办法是要确定一个目标。你的目标可以是为了丰富你的Web页的内容，将你已经拥有的程序转换成Perl，或者为了满足一种无事可做时的好奇心。究竟出于何种目的，这无关紧要。

现在，假如你确定了一个目标，并且具备一定思考能力，那么本书对你有何用处呢？

本书能够为你提供Perl编程语言的基础知识。你能够学到为从事某些工作所必备的Perl知识。你不会被淹没在一些细节之中，那些细节虽然有趣，但只是用于引导你的学习，或许将来再也不会使用它们。本书中的每个新概念都配有大量的工作代码示例。你可以随意翻阅本书，查看你需要的内容。

那么我们究竟为什么要学习Perl呢？从事某种编程开发的所有公司几乎都使用Perl。在金融业、制造业、遗传学、军事和人类知道的所有其他应用领域中，我们都能够找到Perl的踪迹。当然，Perl毫无疑问应用于Internet和World Wide Web。Perl不会很快过时，因此学会Perl的使用技巧是一种非常值得的智力投资，在今后几年中你一定会得到应有的回报。

运用Perl，你将能够使用少量的代码来创建非常出色的Perl程序。使用少量的Perl，将其他应用程序、语言和技术结合起来，你将立即能够创建非常有用的Perl程序。

如何使用本书

本书分为24学时，大约1个小时可以学习一个学时内容。你可以用一天功夫（如果你不吃不睡的话）浏览全书的内容，也可以根据你的时间和自己的节奏浏览各个学时的内容。

在每个学时的结尾处，你可以完成一组新的任务。每个学时的课程都包含该编程语言的各个特性及其作用的说明。此外，每个学时还为你提供了一个自己进行编程练习的机会，你只要按照书上给出的步骤进行操作即可。

本书使用的一些约定

本书使用了一些全书统一的约定，它们是：

- 每个学时的开始是这个学时学习内容的概述。
- 每个学时的结尾处是一个课时小结和一系列常见问题的问答，你可以从中获得有关问题的答案。

此外，本书中还会出现下列图标：



提供书中讲述的内容的补充说明。



提供完成操作任务的捷径和提示。



提醒你使用Perl时可能遇到的困难和解决办法。

第一部分

Perl 基础

- 第1学时 Perl 入门
- 第2学时 Perl 的基本构件：数字和字符串
- 第3学时 控制程序流
- 第4学时 基本构件的堆栈：列表与数组
- 第5学时 进行文件操作
- 第6学时 模式匹配
- 第7学时 哈希结构
- 第8学时 函数

第1学时 Perl入门

Perl是一种通用编程语言。凡是其他编程语言能够使用的地方，都有它的用武之地。在各行各业中，它已经被用于你能够想像到的各种各样的任务的处理。它已经用于股票市场、产品制造、产品设计、客户支持、质量控制、千年虫测试、系统编程、工资处理和库存管理等各个领域，当然还有Web。

Perl的用途之所以如此广泛，原因是Perl被称为是一种“胶水语言”。所谓胶水语言，也就是说它是可以用来将许多元素连接在一起的语言。你可能不想用Perl来编写一个文字处理程序（尽管可以编写这样的程序），因为现在已经有许多非常出色的文字处理程序了。用Perl来编写数据库、电子表格、操作系统或者特性完善的Web服务程序也不是聪明之举，不过编写这些程序是完全可以做到的。

Perl真正擅长的是将这些程序连接在一起。Perl能够利用你的数据库，将它转换成一个具有电子表格特性的文件，并且在你进行文件的处理时，根据需要对数据进行调整。Perl也能够利用文字处理文档，将它们转换成HTML文档，以便在Web上显示。

由于Perl是一种“胶水语言”，能够将许多元素连接在一起，因此它具有极强的适应性。它至少能够在二十几种操作系统下运行，甚至能够在更多的操作系统下运行。Perl的编程样式非常灵活，因此可以用许多不同的方法来做同一件事情。你编写的Perl程序看上去可能与我的程序毫无共同之处，但是如果它们运行起来，却没有任何问题。必要的时候，Perl可以是一种非常严谨的语言，而对于编程新手来说，如果你愿意，它又可以是一种非常随意的语言，这完全可以根据你的需要来定。

下面让我们来澄清一些基本概念。这个编程语言的名字是Perl。运行你的程序的程序（即解释程序）的名字是perl。对你来说它们之间的差别通常并不十分重要，不过当你试图启动你的程序时，情况就不一样了，那时它的名字总是perl。有时，你会看到Perl被写成了PERL，这可能是因为Perl的名字是Practical Extraction and Report Language缩略而来的。现在已经没有人再说PERL，因为这个名字显得太一本正经了。Perl这个名字则比较随便。



Perl的许多特性是从其他语言中借来的。这种借用曾经在早期导致Perl成为另一个用语Pathologically Eclectic Rubbish Lister的缩写。

本学时介绍的内容包括：

- 安装Perl。
- 访问Perl的内部文档。
- 编写你的第一个Perl脚本程序。

1.1 安装Perl

若要使用Perl，首先必须安装Perl。Perl的安装是非常容易的，并且不会出错。实际上，作为安装步骤的一部分，Perl应该进行自我测试，以确保它安装成功。安装的操作过程可以有

很大的差别，这要根据你运行的操作系统而定。因此，为了使安装操作能够进行下去，首先要确定你运行的是什么操作系统，然后再进行下一步工作。

1.1.1 等一等，也许你已经安装了Perl

当你着手在你的系统上安装Perl之前，应该检查一下是否已经安装了Perl。有些UNIX的供应商已经为操作系统配备了Perl。Windows NT也将Perl作为Windows NT的资源工具包(Resource Kit)的组成部分提供给客户。若要查看你的操作系统上是否已经安装了Perl，你需要获得一个命令提示。

在UNIX系统下，只需要登录到该系统中。如果你拥有一个图形操作环境，需要打开终端窗口。当已经登录或者打开终端窗口后，你会看到下面这个提示符：

```
$
```

这个提示符也可能是%，也可能是bash%，无论什么提示符，它都称为shell提示符或命令提示符。在本书的头几个学时中，你将需要使用这个提示符与Perl进行交互操作。

若要了解你的操作系统上是否已经安装了Perl，请键入下面这行命令（不要键入\$提示符）：

```
$ perl -v
```

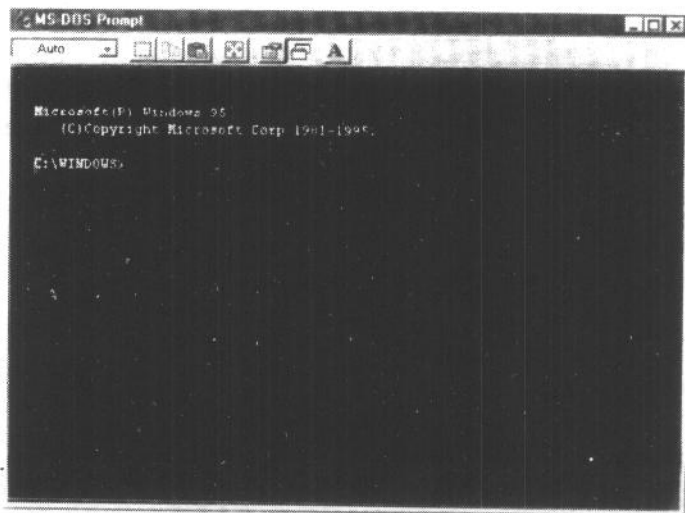
这时系统可能显示一个出错消息，如command not found（命令没有找到），也可能Perl作出响应，输出它的版本号。如果Perl输出它的版本号，那么就表示它已经安装好了，你就不需要重新进行安装。



报告的版本号至少应该是5，也许是5.004、5.005、5.6等，不能小于这些数字。如果Perl的版本号是4.x，那么你必须安装一个新拷贝。Perl 4这个版本太老，错误很多，而且不再能够得到维护，本书中只有很少的示例程序能在Perl 4下运行。在撰写本书时，5.005是Perl的当前版本，5.6在1999年底推出。

如果你拥有一台运行Windows操作系统的计算机，要想查看是否安装了Perl，必须显示图1-1所示的MS-DOS提示符。

图1-1 可以在这个DOS提示符下查看Perl的版本



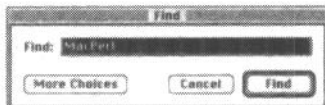
在这个提示符下，键入下面这个命令（不要重复键入提示符）：

```
C:\> perl -v
```

如果Perl已经安装，它就会显示版本号。正如上面的警告中所说的那样，它的版本号至少必须是5。如果DOS回答说Bad command or file name（命令或文件名不正确），那么你就应该安装Perl。

在Macintosh计算机上，你可以像图1-2所示的那样，运行File Find命令（Command-f），在Find的命令框中键入‘MacPerl’，来查看是否已经安装了Perl。如果找到了该应用程序，那么将它打开，观察Apple菜单下面的“About MacPerl”选项。你至少应该拥有Version 5.2.0 Patchlevel 5.004这个版本号，否则就应该安装MacPerl的新版本。

图1-2 查看Macintosh计算机上的Perl



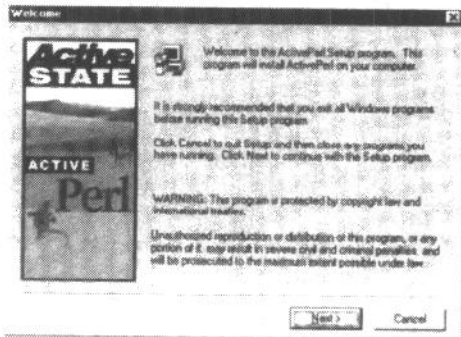
1.1.2 在Windows 95/98/NT上安装Perl

若要在Windows下安装Perl，请记住，你可以像许多其他情况中那样，既可以使用捷径，也可以使用比较笨的办法。如果你对开发环境中需要的C编译器和各种工具（比如Makefile、shell等）非常熟悉，那么可以从头开始安装你自己的Perl。可以随意查看、修改和改变Perl解释程序的源代码，使之适合你的需要。详细情况请参见第16学时（Perl程序的开发界）的内容。想要在Windows下从头安装Perl并不容易，对于大多数人来说，这样做并不值得。

安装Perl时使用的捷径确实是非常容易的。ActiveState Tool公司提供了一个自行安装Perl的工具，安装工作就像安装其他任何Windows应用程序一样，如图1-3所示。这个Perl是在ActiveState Community License（ActiveState团体许可证）下提供的，你应该阅读该许可证的有关说明。该公司的URL是<http://www.ActiveState.com>。

这个Perl与你自己建立的Perl是完全相同的东西。ActiveState公司只是为你做了最困难的那部分工作，并且用有关安装程序将它包装了起来。如果你需要的话，ActiveState公司还为Perl提供了商业上的支持，并且提供了一些附加产品，如调试程序和其他开发工具及文件库。

图1-3 用ActiveState提供的工具在Windows下安装Perl



本书的光盘上包含了ActiveState公司的Perl产品的拷贝，你可以直接使用这个光盘来安装Perl，也可以通过ActiveState公司的Web站点来获取最新版本的Perl。

1.1.3 在UNIX上安装Perl

若要在UNIX上安装Perl, 需要具备两个条件。首先, 需要一个Perl的源模块包的拷贝。你始终都可以从<http://www.perl.com>的Downloads区域下载它的最新版本。可以从那里找到多个版本, 不过你需要的版本总是带有“Stable”或“Production”标号。还需要一个ANSI C编译器。如果不知道这个编译器的作用, 也不必担心。Perl的配置程序能够选定一个, 如果没有这个编译器, 你可以安装一个预安装版本, 这在本书的结尾处将要介绍。



如果你的UNIX配有一个用于安装预安装软件包的系统, 你就能够安装一个预安装的Perl版本。Linux、Solaris、AIX和其他UNIX系统均配有已经捆绑的预安装Perl版本, 它们的安装非常容易。请查看有关资料, 以了解何处能够得到这些软件包。

当你拥有Perl的源模块包后(它的文件名类似Stable.tar.gz), 必须对它进行拆包, 然后进行安装。若要进行操作, 请输入下面的命令:

```
$ gunzip stable.tar.gz
$ tar xf stable.tar
```

这两个命令的运行需要花费一定的时间。如果没有gunzip解压缩程序, 可以从<http://www.gnu.org>下载一个拷贝。该程序包称为gzip。当你完成所有的拆包操作后, 就会看到一个提示符, 然后键入下面的命令:

```
$ sh Configure
```

这时Configure程序就开始运行, 并且问你一系列的问题。如果大部分问题你不知道如何回答, 这没有关系, 你只需要按Enter键即可。默认的答案通常是最好的答案。Perl几乎能够在任何UNIX系统上安装而不会出现任何问题。当所有这些操作完成时, 键入下面的命令:

```
$ make
```

Perl的安装需要花费相当长的时间, 你可以乘此机会喝点儿咖啡。如果你的系统运行速度比较慢, 你可以利用这个时间用午餐。当安装完成时, 再键入下面这两个命令:

```
$ make test
# make install
```

make test这个命令用于确保Perl的安装百分之百正确并且使之可以准备运行。若要运行make install命令, 你必须以一个根用户的身份进行登录(正因为这个原因, 所以使用提示符#, 这是根用户的提示符), 因为它需要将Perl安装到系统目录中去。

当make install运行正确时, 你可以测试Perl的安装情况, 方法是再次键入下面的命令:

```
$ perl -v
```

如果这个命令运行正确, 那么祝贺你安装成功了!



在UNIX下安装Perl时使用的源模块拷贝位于本书所附的光盘上。可以直接从这里拷贝这个安装模块包, 也可以从<http://www.perl.com>那里获得Perl的最新版本。

1.1.4 在Macintosh系统上安装Perl

Macintosh Perl的最新版本称为MacPerl，可以从CPAN端口目录下获得该版本。你必须访问<http://www.perl.com/CPAN/ports/mac>站点，从那里下载安装文件。你应该从该目录下下载MacPerl appl.bin的最新版本，安装时，请使用StuffIt Expander，从下载文件中取出MacPerl安装程序，然后运行该安装程序。

当完成安装后，你可能想要为Perl文档的读者安装一个帮助程序Shuck，它是与MacPerl一道安装的。MacOS 8的用户可以通过Internet控制面板进行安装，方法是打开Advanced->File Mapping，给Shuck应用程序添加文件扩展名映像.pod。这样就可以更加容易地访问该文档了。还可以给MacPerl应用程序建立.ph、.pl、.plx、.pm、.cgi和.xs（这些都是Perl使用的扩展名）等文件映像，请务必将文件类型设置为‘TEXT’。

MacOS 7的用户必须使用InternetConfig实用程序来进行类似的映射操作。在InternetConfig中，选定Helpers，为pod添加新的帮助应用程序shuck。另外，还要将前面提到的其他扩展名的帮助文件添加给MacPerl应用程序。



本书所附的光盘上包含了一个MacPerl的安装软件包的拷贝。你可以直接用光盘进行安装，也可以访问<http://www.perl.com/CPAN/ports/mac>站点，从那里获得一个最新的拷贝。

1.2 文档资料

这个问题非常重要，因此你应该格外注意。每安装一个Perl，你就会得到一份完整的Perl语言和解释程序的当前文档资料的拷贝。

是的，安装软件包配有用于Perl的整套文档资料，你可以免费获得这套资料。Perl 5.005版包含的资料超过1700页。这些资料包括参考资料、培训资料、FAQ、历史资料，甚至是关于Perl内部情况的说明。

可以使用各种不同的方法来访问这些文档资料。在Windows和UNIX系统上，与Perl一道安装了一个称为perldoc的实用程序。你可以使用perldoc程序来搜索这些文档资料，为手册提供格式化输出。若要运行perldoc程序，你必须处在一个命令提示符下。下面这个例子使用的是UNIX提示符，不过在DOS命令提示符下也可以：

```
$ perldoc perl
```

```
PERL(1)                13/Oct/98 (perl 5.005, patch 02)                PERL(1)
```

```
NAME
```

```
perl - Practical Extraction and Report Language
```

```
SYNOPSIS
```

```
perl [ -sTuU ]          [ -hv ] [ -V[:configvar] ]
    [ -cw ] [ -d[:debugger] ] [ -D[number/list] ]
    [ -pna ] [ -Fpattern ] [ -l[octal] ] [ -O[octal] ]
    [ -Idir ] [ -m[-]module ] [ -M[-]'module...' ]
    [ -P ]           [ -S ]           [ -x[dir] ]
    [ -i[extension] ]           [ -e 'command' ] [ --
    ] [ programfile ] [ argument ] ...
```

For ease of access, the Perl manual has been split up into a number of sections:

:

手册的各个部分可以分成不同的节，其名字可以是perlfunc（Perl函数）、perlop（Perl运算符）和perlfaq（Perl FAQ）等。若要访问perlfunc手册页，可以输入命令perldoc perlfunc。手册的所有部分的名字都在perldoc perl手册页中列出。

若要搜索手册，查找某个函数名，可以运行带有-tf开关的perldoc实用程序。下面这个例子用于查找Perl的print函数的手册页：

```
$ perldoc -tf print
```

FAQ是指关于Perl的常见问题。这些问题是学习Perl的人一再提出的问题。为了节省人们的时间，减少一些麻烦，这些问题集中起来放入称为FAQ的文件中。若要搜索FAQ文件，找出某个关键字，你应该使用-q开关，后随FAQ标题中可能出现的一个单词。例如，如果想要知道有关Perl的支持信息，可以使用下面这个命令：

```
$ perldoc -q support
```

这时，就会显示FAQ问题的条目“Who Supports Perl ? Who develops it? Why is it free ?”（谁支持Perl？谁开发了Perl？它为什么是免费的？）

1.2.1 某些特殊的文档资料举例

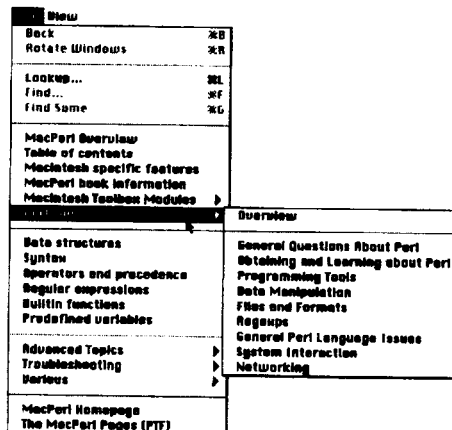
当在UNIX系统上安装Perl时，安装人员会看到一个选项提示，让他按传统的“man”格式来安装手册页。如果安装人员选择yes（是），那么标准Perl文档就转换成man格式，并且存放在一个相应的位置。若要访问Perl文档，既可以使用perldoc程序，也可以像通常在UNIX中那样使用man程序：

```
$ man perl
```

当在Microsoft Windows系统上安装ActiveState公司的Perl产品时，手册页被转换成HTML文档格式，并且可以用Web浏览器进行访问。如果想要阅读手册，请将你的Web浏览器指向本地目录C:\Perl\html，假如你使用标准安装目录的话；如果你不使用标准目录，则使用你选定的目录。

如果是Macintosh系统，MacPerl配有一个称为Shuck的实用程序，它位于MacPerl文件夹中。你可以用它来阅读和搜索Perl文档，如图1-4所示。

图1-4 MacPerl的Shuck文档阅读器



1.2.2 如果无法找到文档该怎么办

如果无法找到你要的文档，那么可能在两个地方出现了错误。首先是没有查找正确的文档位置。也许perldoc实用程序安装到的目录不在你的shell搜索路径上，或者你的搜索路径具有很强的限制性。请反复搜寻perldoc实用程序，将该目录添加到你的shell的PATH环境变量中。

第二个原因是文档被删除了。也许是不小心删除的，也可能是恶意删除的。Perl的安装包括了文档的安装。你不能只安装Perl而不安装文档。如果文档没有了，那么完全可以认为Perl的安装不正确，或者安装以后遭到了破坏。也许你或者系统管理员应该考虑重新安装Perl。文档是Perl开发环境的不可分割的组成部分，如果没有文档，那么Perl的某些部分将无法运行。

如果其他方法都失败了，而你仍然无法得到文档的本地拷贝，那么你可以求助于Web。在Perl的主要销售站点（<http://www.perl.com>）上，可以访问一组标准文档。拥有你的Perl版本配备的实际文档更好，因为它是根据你的特定Perl版本和安装情况裁剪的文档，不过这个在线文档只有在紧急情况下才能使用。

1.3 编写你的第一个Perl程序

若要编写你的Perl程序，需要一个称为文本编辑器的实用程序。使用文本编辑器，可以将不带任何格式的纯文本输入文件中。Microsoft Windows 的Notepad和DOS 的EDIT.EXE均属文本编辑器。在UNIX中，vi、emacs和pico等，都是文本编辑器。你的系统上至少存在其中的一个文本编辑器。在Mac系统下，MacPerl应用程序包含一个基本的文本编辑器，若要打开一个新程序，请在File菜单下选定New。

不应该使用文字处理程序来键入你的Perl程序。文字处理程序，比如Microsoft的Word、Wordpad和WordPerfect等，在保存文档时，会将格式化代码嵌入文档之中，即使文档并不包含任何格式。这些格式化代码会使Perl感到莫名其妙，你的程序将无法正确运行。如果需要使用文字处理程序，那么务必将你的程序保存为纯文本文件。

1.3.1 键入程序

打开文本编辑器，正确无误地键入下面这个Perl程序：

```
#!/usr/bin/perl

print "Hello, World!\n";
```

#!/ 这个程序行应该是文件中的第一行。

当将该程序键入你的文本编辑器后，将它保存在名字为hello的文件中。不需要在这个文件名上使用扩展名，但是你加不加扩展名，Perl都无所谓。有些Windows和Macintosh实用程序使用扩展名来指明该文件是什么类型的文件。如果需要或者想要使用扩展名，那么常用的扩展名是.pl或.plx，因此你可以使用hello.pl这样的文件名。

1.3.2 运行程序

这时，若要运行该程序，必须进入到一个命令提示符下。在UNIX中，请打开一个终端窗口，或者登录到系统中。在Microsoft Windows计算机上，显示一个MS-DOS提示符。还应该

使用你的shell的cd命令，转到你存放hello程序的目录中。

当显示该提示符时，键入下面这个命令。（下面显示的是DOS提示符，UNIX提示符稍有不同。）

```
C:\PROGRAMS> perl hello
```

如果一切进行正常，Perl应该显示下面这行消息：

```
Hello, world!
```

如果你键入的这个命令运行正确，那么祝贺你的程序运行成功了！请记住如何运行这个程序，因为这也是你启动本书其余章节中的程序的方法。（也可以使用其他一些方法，下面很快就要介绍这些方法。）

如果该命令不能运行，请进行下列检查：

- 如果看到出错消息Bad command or file name或者perl : command not found，那么表示perl程序不在你的执行路径上。必须确定perl程序究竟安装在什么地方，并将该目录添加给你的shell中的PATH变量。
- 如果看到出错消息Can't open perl script hello : A file or directory does not exist（无法打开Perl脚本hello：文件或目录不存在），那么你可能没有进入早些时候保存hello文件所在的这个目录，也许你将该文件保存到另一个目录中了。
- 如果看到syntax error这样的错误，那么Perl能够正常启动运行，但是无法确定hello文件中究竟有什么。也许键入的文件内容有错误，也可能使用了一个文字处理程序，它将格式应用到保存的文件中了。可以使用UNIX的cat命令，或者DOS的type命令，来核定文件中的内容。如果键入的内容有误，你应该对所有内容进行检查，引号和所有标点符号都很重要。

如果使用MacPerl，只需要从Script菜单中选定Run“hello”，运行你的第一个perl程序。如果没有使用MacPerl的内置编辑器来编写你的程序，那么请从File菜单中选定Open命令，打开MacPerl中你的程序，然后选定Run。

1.3.3 程序正确将会发生什么情况

当键入命令perl hello时，你的计算机上一个名叫perl的程序就启动运行。该程序称为perl解释程序。这个perl解释程序是Perl的核心和灵魂。它的作用是取出对它赋予的文件(这里是文件hello)，找出里面的程序，并运行该程序。

所谓“运行该程序”，是指首先要检查构成Perl程序的语句、函数、运算符、数学算法和其他所有元素，以确保句法正确，然后每次运行一个语句。

当perl解释程序完成了从磁盘中读取你的程序的全部操作后，它即开始运行该程序，并且继续运行，直到整个程序运行完成。当它完成程序的运行时，perl解释程序退出，将控制权重新交给你的操作系统。

现在让我们来看一看hello程序是如何“运行”的。

1.3.4 Perl程序的具体运行过程

Hello程序的第一行是：

```
#!/usr/bin/perl
```

对于Perl来说，代码行上的 `#` 符号后面的一切均被视为注释行。注释是指Perl将对它加以忽略的一些东西。在某些情况下，程序的第一行上的 `#!` 是不同的。它后面的路径名 `/usr/bin/perl` 是到达perl解释程序的路径。如果UNIX程序有一行是以 `#!` 开头，后随一个解释程序的路径，那么UNIX就知道这是个程序，并且可以按照名字来运行。关于如何运行程序的说明，请参见本学时结尾处的“专家答疑”。

有些能够运行Perl程序的Web服务器，比如Apache，也对 `#!` 这个程序行非常注意，并且能够在没有perl命令的显式说明下运行该程序。

现在我们正好是将 `#!` 视为一个注释行。

下一个程序行是：

```
print "Hello,World!\n";
```

这里包含了许多内容。这一行构成了一个Perl语句，它为Perl标明了一项需要执行的操作。

首先，这一行包含一个函数，称为print。这个print函数取出它后面的所有内容，并默认显示在屏幕上。print函数的输出结果，是直到分号（`;`）前的那部分内容。

Perl中的分号是个语句分隔符。你应该在Perl程序中的各个语句之间放一个分隔符，以便显示一个语句的结束和另一个语句的开始。

在这个例子中，print函数显示了短语Hello,World!。程序行结尾处的`\n`告诉Perl将一个新的空行插入到它输出的短语的后面。短语前后的引号和`\n`告诉Perl，这是个文字串，不是另一个函数。下一个学时将要非常详细地介绍字符串的内容。

1.3.5 必须知道的一些情况

Perl被人们称为是一种自由格式的编程语言。这意味着Perl语句在编写的时候并不是非常严格。可以将空格、制表符，甚至回车符（它们称为白空间）插入Perl语句中的任何位置，这实际上都没有关系。

但是有些位置不能随意插入白空间，这些位置是应该加以限制的位置。例如，不能在函数名的中间插入空格，`pr int`是个无效函数。另外不能将空格插入数字中，比如25 61这个数字是不行的。像“Hello World!”这样的文字串中的白空间当然可以显示为白空间。几乎任何其他地方它都是有效的。你可以编写类似下面这样的Perl程序示例：

```
#!/usr/bin/perl

print
    "Hello, World!\n"
;
```

这个程序在功能上与原先的程序是相同的。Perl语言的这种自由格式特性使你的Perl程序可以具有非常丰富的“样式”。在程序的格式上可以有很大的随意性。不过请记住，总有一天其他用户会查看你的程序的，因此你不能让他们看不懂。

本书中的程序所用的样式是相当保守的。有时为了清楚起见，或者为了节省空间，语句被分成了若干行，因为Perl的语句可能非常长。Perl的文档资料甚至提供了一个建议性的样式指南，可以浏览该文档，以便了解有关的建议。可以搜索名字为perlstyle的文档。



Perl程序中的样式有时可能非常特殊。有的Perl程序可以写成诗歌，甚至俳句，这些都是有效的。有些令人难忘的Perl程序看上去就像是图画，不过它们仍然能够做一些有用的工作。《Perl Journal》（网址是<http://www.tPj.com>）每年都要举行一次编写样式特殊的Perl程序的比赛，比赛的名字是Obfuscated Perl Contest。你不应该采取这些程序项目的样式。

1.4 课时小结

在本学时中，我们学习了一些关于Perl和Perl是如何运行的知识。随着你阅读本书其他章节的内容，你会不断增加对它的了解。还学习了如何在你的系统上安装Perl，并且如何来检验它是否运行正确以及它的所有文档资料是否已经安装到位。最后，键入并且运行了你的第一个Perl程序。接着分析了这个程序，并且进一步学习了Perl如何运行的一些知识。

1.5 课外作业

1.5.1 专家答疑

问题：Perl运行的那些东西是称为Perl脚本还是Perl程序？

解答：名字实际上并不重要。传统上，程序被编译成机器语言并且按照机器语言的形式来存放，机器语言是可以多次运行的。而脚本可以被放入一个外部程序中，每当脚本运行时，外部程序就将脚本转换成一些操作。Perl的发明人Larry Wall曾经说过：“脚本是你为操作人员提供的东西，而程序则是你提供给用户的東西。”你可以随意称呼它们。在本书的其余章节中，将它们称为Perl程序，如果你学习成绩很好的话，那么你就可以称为一名Perl程序员。

问题：是否必须键入本书中的一些程序清单？有些程序清单非常长。

解答：本书中的所有程序清单和程序示例，以及这些程序需要的数据文件等，都在本书所附的光盘上。

问题：在“运行程序”这一节中，讲到在UNIX下有一种比较简便的方法可以用来运行Perl程序。究竟如何运行呢？

解答：首先，必须确保程序的 `#!` 行正确无误，同时路径名确实指向一个perl解释程序；`/usr/bin/perl`是它的通常位置，在有些机器上它的位置是`/usr/local/bin/perl`。接着，必须使用`chmod`命令，使该程序能够执行。如果是hello程序，那么UNIX的shell命令是`chmod 755 hello`。这项操作完成后，你可以键入`hello`或`./hello`，运行该Perl程序。应该说明的是，在UNIX下，不要将你的程序命名为“test”。UNIX的shell有一个命令的名字是test，当错误地运行test命令时，就会带来很大的麻烦。关于不能使用的其他程序名，请参见你的shell的文档资料。

另外，如果使用光盘上的程序清单，必须修改 `#!` 行，使之与你的系统上的Perl位置相一致，否则，你将必须从命令提示符下键入`perl programname`运行你的程序。

1.5.2 思考题

1) Perl是编程语言的名字；perl则

a. 也是该语言的名字。