

软件研发精品译丛

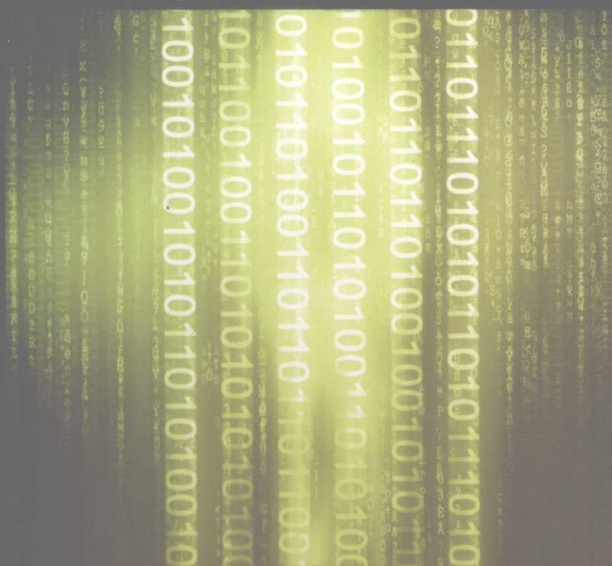
RUBY ON RAILS

for PHP and Java Developers

Ruby on Rails 实践之路

——写给PHP和Java开发者的书

Deepak Vohra 著
祁飞 何箐 译



科学出版社
www.sciencep.com

Ruby on Rails 实践之路

——写给 PHP 和 Java 开发者的书

Deepak Vohra 著
祁 飞 何 菁 译

科 学 出 版 社

北 京

图字:01-2008-3374号

内 容 简 介

Ruby on Rails 是基于 MVC 模式的 Web 框架,用于开发基于数据库的 Web 应用。Ruby on Rails 中内含了所需的 Web 服务器 WEBrick。该框架配置的数据库除了缺省的 MySQL 外,还可以是 Oracle、SQL Server 等其他数据库。本书覆盖了使用 Ruby on Rails 开发 Web 应用的各项技术,包括 Ajax、目录服务以及 Web 服务等。书中将 Ruby on Rails 与主流编程语言 Java 和开发 Web 应用最普遍的脚本语言 PHP 进行了比较,易于读者理解和掌握。书中提供的示例应用程序修改后读者可以直接用于实际的项目开发活动中。

本书适合高等院校软件工程专业师生以及 Web 应用开发人员、IT 架构师、项目经理、系统管理员等技术人员阅读参考。

Translation from English edition:

Ruby on Rails for PHP and Java Developers by Deepak Vohra

Copyright © Springer-Verlag Berlin Heidelberg 2007

Springer is a part of Springer Science + Business Media

All Rights Reserved

图书在版编目(CIP)数据

Ruby on Rails 实践之路:写给 PHP 和 Java 开发者的书/(印)沃哈(Vohra, D.)著;祁飞,何箐译. —北京:科学出版社,2010.5

ISBN 978-7-03-027320-8

I. ①R… II. ①沃…②祁…③何… III. ①计算机网络-程序设计
②PHP语言-程序设计③JAVA语言-程序设计 IV. ①TP393.09②TP312

中国版本图书馆 CIP 数据核字(2010)第 073281 号

责任编辑:任 静 王志欣 / 责任校对:邹慧卿

责任印制:赵 博 / 封面设计:无极书装

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

源海印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

*

2010年5月第 一 版 开本: B5 (720×1000)

2010年5月第一次印刷 印张: 20

印数: 1—3 500 字数: 386 000

定价: 48.00 元

(如有印装质量问题, 我社负责调换)

前 言

1995年发布的 Ruby^①是一种开源的编程语言。Ruby在 TIOBE 编程社区索引^②中排名第10。Ruby on Rails^③是2004年发布的一种开源的 Web 框架,因其简单、可维护性好、开发速度快等优点而被广泛采用。计算机世界(Computer-world)网站^④将 Ruby on Rails 列为2007年五大技术之一。

写这本书的动机

Ruby 常被人拿来与 PHP 和 Java 作比较,其中 PHP 是最常用的脚本语言,而 Java 则是最常用的编程语言。在 TIOBE 索引的趋势图上你也许会注意到,自2006年中期开始 Ruby 的使用在增长,而 PHP 和 Java 的使用在降低。O'Reilly Radar^⑤也注意到 Ruby on Rails 使用的增长趋势。客观地讲,Ruby on Rails 在某些方面性能优于 PHP,如:

(1) 与 PHP 相比,Ruby on Rails 面向对象的程度更高,它提供了面向对象语言所有的内在优点,例如简单、模块化、可移植性、可扩展性、维护性以及可复用性。

(2) Ruby on Rails 是 Web 框架,而 PHP 只是一种脚本语言。

(3) Ruby on Rails 自带一个开发所需的 Web 服务器,而开发 PHP 程序时必须另安装 Web 服务器。

(4) Ruby on Rails 实现了模型-视图-控制器(model-view-controller, MVC)这种架构,而 PHP 没有实现 MVC 架构,需要时必须另外实现。

(5) Ruby on Rails 框架提供了创建-读取-修改-删除(create-read-update-delete, CRUD)功能,PHP 并未提供。

虽然 Java 在可伸缩性、安全性、事务管理和开发工具的易获性方面都要优于 Ruby on Rails,但是 Ruby on Rails 在简易度、开发速度和可维护性方面又好于 Java。Ruby on Rails 提供的 Web 框架内部成分之间耦合紧密,该框架由持久化框架、Web 应用程序框架和 Web 服务框架三部分构成。J2EE 应用框架无法做到

① Ruby-<http://www.ruby-lang.org/en/>

② TIOBE Index-<http://www.tiobe.com/tpci.htm>

③ Ruby on Rails-<http://www.rubyonrails.org/>

④ ComputerWorld-<http://computerworld.com/action/article.do?command=viewArticleBasic&articleId=9011969>

⑤ O'Reilly Radar-http://radar.oreilly.com/archives/2006/08/programming_language_trends_1.html

Ruby on Rails 这样的无缝集成,它需要做大量的配置工作。相对于 Java 的变量是静态类型来说,Ruby on Rails 则能做到数据是动态类型的。

目前有许多关于 Ruby on Rails 的著作已经出版,但是它们都没有将 Ruby on Rails 与 PHP 做比较,而将 Ruby on Rails 与 Java 做比较的著作则更是稀少。

谁该阅读这本书?

书中相关的大多数章节都会涉及与 PHP 和 Java 的比较。但是本书读者不需要一定是 PHP 或 Java 开发者。也许你正在使用另一种脚本语言,也许 Ruby on Rails 是你正接触的基于 Web 框架的第一种脚本语言。如果使用过 Ajax,你对服务器端的需求会比较熟悉:可能基于 PHP,也可能基于 Java,还可能基于 .NET。Ruby on Rails 支持 Ajax,根据 Ajaxian.com 网站的调查^①,14% 的 Ajax 开发者使用 Rails 作为 Ajax 平台。如果你一直在使用 PHP 或 Java 类库来开发 PDF 和 Excel 报表,Ruby on Rails 则可以为你提供 Ruby 自己的好工具来生成 PDF 和 Excel 报表。如果你是一个 Web 开发者,常常使用 PHP 或者 Java 来创建、读取、修改、删除数据库中的数据项,Ruby on Rails 则可以提供内置的对 CRUD 操作的支持。如果你一直使用 Java 命名和目录接口(Java naming and directory interface,JNDI)或者 PHP 类库来实现目录服务,Ruby on Rails 提供了另一种实现目录服务的途径:创建轻量级目录访问协议(lightweight directory access protocol,LDAP)目录服务。

本书的目的不在于引导读者学习 PHP 或者 Java。本书的大部分章节都包含有一个 Ruby 与 PHP、Java 的简要比较。读者应当注意到 Ruby 与 PHP、Java 之间的相似之处,因为它们同是面向对象语言。如果读者不熟悉 PHP 或者 Java,则可以跳过这些有关语言之间进行比较的章节内容。

本书框架结构

第 1 章中介绍了 Ruby 编程语言。我们安装 Ruby,并使用 IRB(interactive Ruby shell)对 Ruby 语言的语法进行讨论。本章最后将 Ruby 与 PHP、Java 做了对比。

第 2 章中介绍了 Rails 框架。我们安装 Rails 框架,并对组成 Rails 框架的各个不同部分进行讨论。我们还讨论 Rails 请求是如何被路由的,以及如何使用帮助器模块(helper)、布局和样式表。我们为 Rails 框架配置了 MySQL 数据库。本章最后将 Rails 与 PHP、Java 做了对比。

^① Ajaxian.com Survey-<http://ajaxian.com/archives/ajaxiancom-2006-survey-results>

第 3 章中讨论了 Ruby on Rails 提供的 CRUD 操作功能。我们在 MySQL 数据库中创建一个数据库表,同时也讨论了在 Rails 中配置 Oracle 和 SQL Server 数据库的方法。我们还构建了一个创建、读取、修改与删除目录项的 CRUD 应用。

第 4 章介绍了 Ajax,并讨论了 Rails 框架中对 Ajax 的支持。我们还创建了一个数据库查找应用程序,用来列出指定满足条件的目录项。

第 5 章讨论了用 Ruby on Rails 创建 PDF 和 Excel 电子表格的报表生成过程。并将该过程与使用 PHP、Java 类库生成报表的过程做了对比。

第 6 章讨论了 Ruby 中处理 XML 的好工具。我们不仅使用包管理器 Ruby-Gems 的 Builder 2.0.0 创建 XML 文档,我们还从 Oracle 数据库的表中创建 XML 文档。同时,我们也讨论了如何使用 Ruby 中的 XML 处理器 REXML 对 XML 文档进行解析。

第 7 章讨论了 PHP on Trax,它是基于 Ruby on Rails 之上的用于 PHP 的 Web 应用程序与持久化框架。我们创建了一个与第 3 章类似的 CRUD 应用程序,但是没有使用 Ruby。

第 8 章讨论了使用 Ruby on Rails 创建目录服务的方法。同时,我们还讨论了几种常用目录服务器的安装过程。

第 9 章讨论了 Ruby on Rails 中对 Web Service 的支持。我们讨论了各种不同的分发模式和协议客户端,并将使用 Ruby on Rails 创建 Web Service 的过程与使用 PHP、Java 创建 Web Service 的过程进行了对比。

第 10 章讨论了 Eclipse 中支持 Ruby on Rails 的相关插件。我们解释了 Ruby 开发工具(Ruby development tools, RDT)和 RadRails 插件的安装过程和使用方法。

第 11 章讨论了 Rails 测试。没有测试的开发过程是不完整的。我们讨论了针对 Rails 模型的单元测试和针对 Rails 中各控制器(controller)的功能测试,此外还讨论了测试用的样本数据——夹具(fixture)。

第 12 章讨论了 Ruby on Rails 中软件成品(production)发布方面的内容。我们介绍了在 Apache2 和 FastCGI 上部署 Rails 应用程序的过程,并讨论了可以改善性能的 Ruby on Rails 最佳实践方法,以及把 Rails 应用程序托管到 Web 托管主机上的方法。

阅读本书所需的知识准备

阅读本书,读者需要熟悉 PHP、Java,以便能够将 Ruby on Rails 与 PHP、Java 进行比较。读者也应该通晓面向对象的相关概念,如类、方法和继承等。虽然书中在每一章最前面都有“本章简介”,但还需要读者了解 Ajax、目录服务和 Web 服务等相关概念。

致谢

在这里作者要感谢 Springer 出版社的计算机科学执行编辑 Engesser。同时,要感谢 Springer 的项目经理 Fischer 和 LE-TeX Jelonek 的发行编辑 Reinfarth。

关于作者

Deepak Vohra 是 Sun 认证 Java 程序员与 Web 构件开发人员,拥有 South Illinois 大学 Carbondale 分校机械工程专业的理学硕士学位,他同时还是 Oracle 认证专员。此外,他是 Manning 出版社的技术编辑,曾经负责过 *The Prototype and Scriptaculous in Action* 一书的编辑工作。

信息源

Ruby on Rails 网站:<http://www.rubyonrails.org/>

Ruby 网站:<http://www.ruby-lang.org/en/>

Rails 框架的 API:<http://api.rubyonrails.org/>

目 录

前言

1 Ruby	1
1.1 本章简介	1
1.2 安装 Ruby	1
1.3 创建一个 Ruby 应用程序	3
1.4 标识符与注释	4
1.5 字符串	5
1.6 数组、哈希表与范围	7
1.7 变量、常量与运算符	9
1.8 类	13
1.9 方法	15
1.10 Proc 对象和代码块 block	20
1.11 控制结构与迭代子 Iterator	24
1.12 异常处理	29
1.13 模块	31
1.14 Ruby 与 PHP 的比较	32
1.15 Ruby 与 Java 的比较	32
1.16 本章小结	33
2 Rails 框架	34
2.1 本章简介	34
2.2 Rails 概述	34
2.3 活动记录	35
2.4 行为控制器	39
2.5 行为视图	41
2.6 Ruby on Rails 命令	43
2.7 安装 Rails	44
2.8 开发一个 Rails 应用程序	45
2.9 Rails 路由	46
2.10 样式表	46
2.11 帮助器模块 Helper	47

2.12	布局 Layout	47
2.13	局部视图 Partial	49
2.14	Rails 框架实例	51
2.15	为 Rails 配置 MySQL 数据库	55
2.16	Rails 与 PHP 的比较	56
2.17	Rails 与 Java 的比较	56
2.18	本章小结	57
3	Rails 中的 CRUD 操作	58
3.1	本章简介	58
3.2	脚手架	58
3.3	用 PHP 实现 CRUD	59
3.4	用 Java 实现 CRUD	61
3.5	创建一个 Rails 应用程序	61
3.6	创建数据库中的表	62
3.7	迁移 Migration	63
3.8	用迁移创建数据库的 catalogs 表	67
3.9	配置 Oracle 数据库	68
3.10	配置 SQL Server 2005 数据库	70
3.11	开发一个 CRUD 应用程序	71
3.12	Ajax 脚手架	78
3.13	合法性检验	82
3.14	本章小结	85
4	Rails 中的 Ajax	86
4.1	本章简介	86
4.2	Ajax 概述	86
4.3	XMLHttpRequest 概述	87
4.4	创建一个 XMLHttpRequest 对象	88
4.5	打开一个 HTTP 请求	89
4.6	发送一个 HTTP 请求	89
4.7	处理一个 HTTP 响应	90
4.8	在 PHP 中使用 Ajax	90
4.9	在 Java 中使用 Ajax	91
4.10	Rails 对 Ajax 的支持	91
4.11	创建一个 Ruby on Rails 应用程序	95
4.12	创建一个数据库表	96

4.13	发送一个请求	99
4.14	处理一个请求	102
4.15	处理一个响应	108
4.16	本章小结	113
5	创建 PDF 和 Excel 报表	114
5.1	本章简介	114
5.2	用 PHP 创建 PDF 文档	115
5.3	用 Java 创建 PDF 文档	115
5.4	用 Ruby on Rails 创建 PDF 文件	116
5.5	在 PDF 中创建一个表格	119
5.6	用 PHP 创建电子表格	124
5.7	用 Java 创建电子表格	125
5.8	用 Ruby on Rails 创建 Excel 电子表格	126
5.9	用 Ruby 的 Spreadsheet 库创建电子表格	134
5.10	本章小结	138
6	Rails 中的 XML	139
6.1	本章简介	139
6.2	用 PHP 5 处理 XML	139
6.3	用 Java 处理 XML	140
6.4	安装 XML Builder	140
6.5	用 Ruby on Rails 创建 XML 文档	141
6.6	从数据库创建 XML 文档	145
6.7	用 REXML 解析 XML 文档	148
6.8	本章小结	152
7	Rails 框架之上的 PHP 框架	153
7.1	本章简介	153
7.2	安装 PHP	153
7.3	安装 PHPOnTrax	154
7.4	创建一个 Trax 应用程序	155
7.5	创建一个 CRUD 应用程序	158
7.6	本章小结	165
8	Rails 对 LDAP 的支持	166
8.1	本章简介	166
8.2	安装 OpenLDAP	166
8.3	安装 Tivoli Directory Server	171

8.4	安装 Oracle Internet Directory	182
8.5	在 PHP 中使用 LDAP	191
8.6	在 Java 中使用 LDAP	191
8.7	安装 NET::LDAP	193
8.8	创建一个 Rails 应用程序	194
8.9	创建一个目录项	195
8.10	修改一个目录项	200
8.11	查找一个目录项	203
8.12	删除一个目录项	206
8.13	本章小结	212
9	使用 Rails 开发 Web Service	213
9.1	本章简介	213
9.2	用 PHP 实现 Web Service	213
9.3	用 Java 实现 Web Service	214
9.4	用 Ruby on Rails 创建一个 Web Service	216
9.5	Web Service 的 API 类	220
9.6	分发	220
9.7	协议客户端	225
9.8	本章小结	231
10	Eclipse 中的 Ruby on Rails	232
10.1	本章简介	232
10.2	Eclipse 中的 PHP	232
10.3	安装 RDT	232
10.4	在 RDT 中创建一个 Rails 的工程	234
10.5	用 RDT 创建一个数据库表	240
10.6	用 RDT 创建一个 CRUD 应用程序	243
10.7	安装 RadRails	245
10.8	用 RadRails 创建一个 Rails 应用程序	246
10.9	用 RadRails 创建一个 CRUD 应用程序	248
10.10	本章小结	250
11	Rails 测试与夹具	251
11.1	本章简介	251
11.2	PHP 与 Java 中的单元测试	251
11.3	Rails 测试的例子	252
11.4	断言类型	256

11.5	Rails 测试	259
11.6	夹具	260
11.7	单元测试	262
11.8	测试控制器	266
11.9	本章小结	272
12	在产品模式的环境下运行 Rails 应用程序	273
12.1	本章简介	273
12.2	设置成产品模式	273
12.3	Rails 最佳实践方法与性能	273
12.4	在 Apache2 和 FastCGI 上部署应用程序	276
12.5	Rails 应用程序的 Web 主机托管	281
12.6	本章小结	299
索引	300

1 Ruby

1.1 本章简介

Ruby 是一种解释型脚本语言,支持面向对象程序设计。解释型意味着一个 Ruby 应用程序不需要事先编译即可运行。Ruby 中的变量没有类型,可以把任意类型的数据赋值给一个 Ruby 变量。Ruby 中的变量无需事先定义就可使用。Ruby 具有面向对象语言的特征,比如类、继承和方法等。Ruby 中的一切(包括方法、字符串、浮点数和整数等)皆为对象。一个 Ruby 脚本储存在扩展名为 .rb 的文件中,通过 ruby 命令运行。首先,我们需要安装 Ruby。

1.2 安装 Ruby

本节中我们将安装 Ruby 和 RubyGems,其中 RubyGems 是标准的 Ruby 包管理器,可以与 Ruby 应用程序和类库一起使用。安装 Ruby 和 RubyGems 的流程如下:下载 Ruby Windows 安装程序,双击 ruby184-19.exe 程序图标,Ruby 安装向导启动之后,单击 Next,如图 1.1 所示。



图 1.1 Ruby 安装向导

接受使用许可协议,单击 Next。选择默认的组件设置进行安装,其中包括 Ruby-Gems 包管理器和基于 Scintilla^① 的文本编辑器 SciTE,然后单击 Next,如图 1.2 所示。

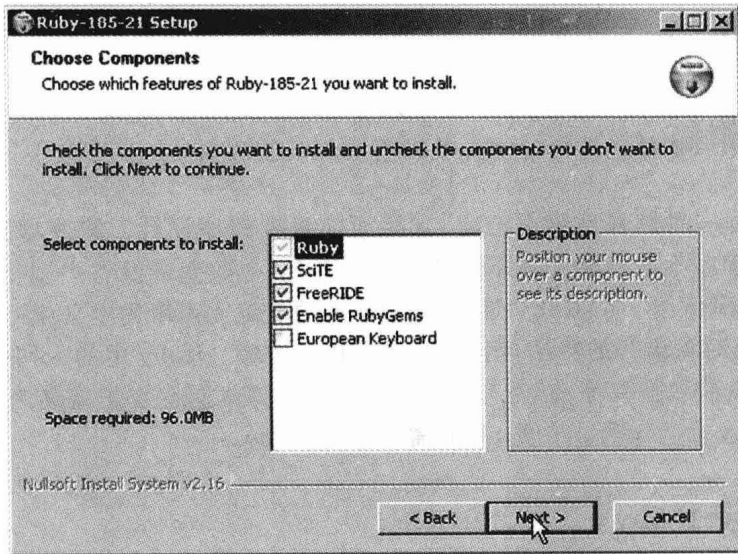


图 1.2 选择待安装的组件

指定安装 Ruby 的目录(默认路径为 c:/ruby),然后单击 Next,如图 1.3 所示。

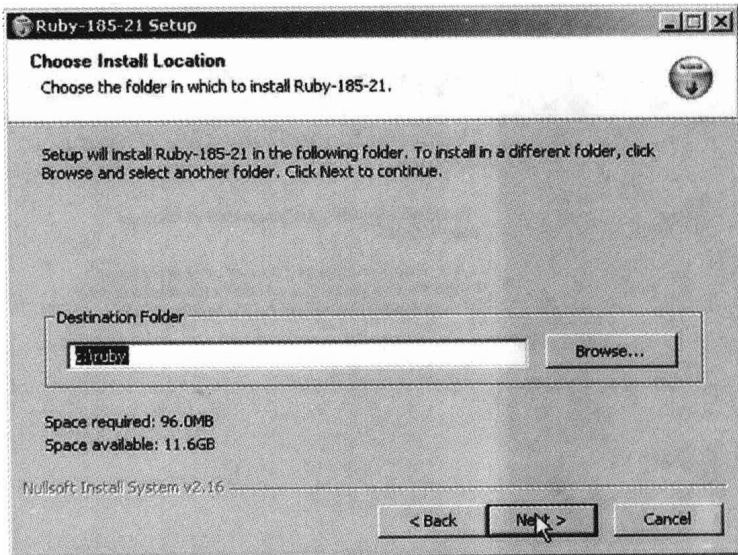


图 1.3 指定安装的文件夹

① Scintilla 是一个开源的编辑组件。

指定一个用于 Windows 开始菜单的文件夹名,单击 Install。Ruby 和 Ruby-Gems 开始安装。单击 Finish 以关闭 Ruby 安装向导。Ruby 安装完后,在系统环境变量 PATH 里添加了目录路径 c:/ruby/bin,在系统环境变量中添加了 RUBYOPT,其值为-rubygems。

1.3 创建一个 Ruby 应用程序

下面,我们来创建一个 Ruby 应用程序。这个例子中,创建的 Ruby 脚本是 helloruby.rb,其中的 Ruby 代码为:

```
puts "Hello Ruby"
```

调用下面的命令来运行该 Ruby 脚本:

```
C: />ruby helloruby.rb
```

该 Ruby 脚本输出如下:

```
Hello Ruby
```

函数 puts 用于打印文本。该函数在每一个文本字符串后面添加一个新行。例如,对上面的 Ruby 脚本进行修改,用字符串分隔符隔开两个词。如下所示:

```
puts "Hello", "Ruby"
```

该 Ruby 脚本输出如下:

```
Hello
```

```
Ruby
```

如果在 Ruby 脚本中没有使用字符串分隔符,则会将字符串连接起来,如下所示:

```
puts "Hello" "Ruby"
```

该 Ruby 脚本输出如下:

```
HelloRuby
```

函数 print 是另一个用于打印字符串的命令。函数 print 和函数 puts 之间的区别在于函数 print 不会在字符串后添加新行,除非脚本中专门声明要加一个新行。

Ruby 也为我们提供了用于运行 Ruby 脚本的交互式 shell。这个交互式 shell 可以用 irb 启动,如图 1.4 所示。



图 1.4 Ruby 交互式 shell

本章中 Ruby 脚本的例子就是在 irb 中运行的。交互式 shell 中运行 Ruby 脚本的方法如下所示:

```
irb(main):001:0> puts "Hello Ruby"
```

从图 1.5 可以看出,运行后的输出结果与运行一个 .rb 脚本文件是相同的。

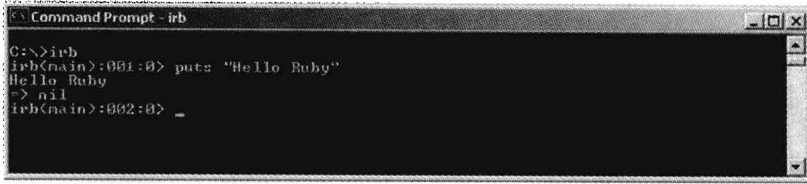


图 1.5 在 irb 中运行一个 Ruby 脚本

Ruby 也提供了函数 gets,用来接收用户输入的字符串。在 Ruby 交互式 shell 中输入 gets 并回车:

```
irb(main):001:0>gets
```

再输入一个字符串,例如“Hello Ruby”并回车,此字符串将被输出,如图 1.6 所示。



图 1.6 函数 gets

1.4 标识符与注释

标识符就是 Ruby 脚本中的变量名、方法名和类名,通过这些名称来区分变量、方法和类。一个 Ruby 标识符以一个英文字母[a-zA-Z]或者一个下划线‘_’开始,后接若干个英文字母、数字或下划线。类名的首字符必须是大写英文字母。标识符不能是保留字。保留字列举如下:

- | | | | | | |
|---------|-------|-------|--------|----------|--------|
| = begin | = end | alias | and | begin | BEGIN |
| break | case | class | def | defined? | do |
| else | elsif | END | end | ensure | false |
| for | if | in | module | next | nil |
| not | or | redo | rescue | retry | return |
| self | super | then | true | undef | unless |
| until | when | while | yield | | |

单行注释以 # 开始,直到本行行尾处结束:

```
# Example of a comment
```

在脚本中的 =begin 和 =end 之间可以嵌入文档,形成多行注释。多行的注释

文档形式如下所示：

```
= begin
  Example Of
  Embedded Documentation
= end
```

1.5 字符串

表示一个字符串时，串的内容既可以用单引号括起来，也可以用双引号括起来。当字符串用单引号括起来时，字符串内容中的单引号可以用转义符\ '表示，反斜杠可以用转义符\\表示。当字符串用双引号括起来时，字符串内容中的双引号可以用转义符\"表示，反斜杠可以用转义符\\表示，还有其他一些转义符：如回退一格(backspace)用转义符\b、回车(return)用转义符\r、换行(newline)用转义符\n、空格(space)用转义符\s、横向制表(tab)用转义符\t。当字符串用双引号括起来时，对标记#{ }中嵌入的字符串表达式还可以进行求值。例如，运行以下 Ruby 脚本：

```
puts "#{ "Hello" + "Ruby" }"
```

则输出为：

```
HelloRuby
```

在标记#{ }中引用的变量必须是预先定义好的。在上面的例子中，运算符+用于连接字符串。运算符*可用于重复字符串，如下所示：

```
puts "Hello Ruby" * 3
```

则此脚本的输出如下：

```
Hello RubyHello RubyHello Ruby
```

Ruby 中的字符其实是整数。字符可以从字符串中提取出来，如下所示：

```
hello = "Hello Ruby"
puts hello[8]
```

字符下标从 0 开始算起。上面脚本的输出为字符‘b’的 ASCII 编码：98。也可以通过指定字符串起始下标和终止下标来提取该字符串中的子串，如下所示：

```
hello = "Hello Ruby"
puts hello[6,10]
```

则此脚本的输出为：

```
Ruby
```

还可以从字符串尾开始来指定字符的下标偏移量，这时下标偏移量前加负号。从字符串尾开始的下标偏移量从 1 开始算起。第二个参数表示子串所包含的字符个数，如下所示：

```
hello = "Hello Ruby"
```