

目录	1
----	---

目录

第一部分 泰山职业技术学院	1
1 信息技术工程系	1
1.1 计算机应用专业	1
2 建筑技术工程系	2
2.1 工程预算专业	2
第二部分 服装职业技术学院	2
3 服装设计系	2
3.1 服装展示专业	2
第三部分 前言部分	2
4 松本行弘的程序世界	3
5 内容提要	4
6 推荐序	5
7 中文版序	7
8 DONE 前言 <2016-08-30 二 14:10>	8

第一部分 泰山职业技术学院

1 信息技术工程系

1.1 计算机应用专业

1. 专业建设方案

中华人民共和国果十周年纪念款 iPhoneX 不仅带来了令人欣喜的全面屏，更是带来了令人纠结的“齐刘海”。追求极致美学的苹果为什么会允许这样“逆天”的设计存在呢？这个“齐刘海”又到底有什么用？今天编辑就来深度解析一下 iPhoneX 的“齐刘海”。

2 建筑技术工程系

2.1 工程预算专业

1. 专业建设方案

服装职业技术学院果十周年纪念款 iPhoneX 不仅带来了令人欣喜的全面屏，更是带来了令人纠结的“齐刘海”。追求极致美学的苹果为什么会允许这样“逆天”的设计存在呢？这个“齐刘海”又到底有什么用？今天编辑就来深度解析一下 iPhoneX 的“齐刘海”。

第二部分 服装职业技术学院

3 服装设计系

3.1 服装展示专业

1. 专业建设方案

第三部分 前言部分

L^AT_EX \

4 松本行弘的程序世界

本书为“Ruby”之父经典力作，展现了大师级的程序思考方式。作者凭借对编程本质的深刻认识和对各种技术优缺点的掌握，阐述了 Ruby 的设计理念，并由此延伸，带领读者了解编程的本质，一窥程序设计的奥秘。本书不是为了介绍某种特定的技术，而是从宏观的角度讨论与编程相关的各种技术。书中第 1 章介绍了作者对编程问题的新思考和新看法，剩下的内容出处《日经 Linux》杂志于 2005 年 5 月到 2009 年 4 月连载的“松本编程模式讲坛”，其中真正涉及“模式”的内容并不多，大量篇幅都用于介绍技术内幕和背景分析等内容，使读者真正了解相关技术的立足点。另外，书中还包含许多以 Ruby、Lisp、Smalltalk、Erlang、JavaScript 等动态语言所写成的范例。

- Ruby 之父佳作，进入不同凡响的程序世界
- 深入剖析程序设计的道与术
- 举一反三，触类旁通

LaTeX

5 内容提要

本书是探索程序设计思想和方法的经典之作。作者从全局的角度，利用大量的程序示例及图表，深刻阐述了 Rub 编程语言的设计理念，并以独特的视角考察了与编程相关的各种技术。阅读本书不仅可以深入了解编程领域各个要素之间的关系，而且能够学到大师的思考方法。本书面向各层次程序设计人员和编程爱好者，也可以供相关技术人员参考。

L^AT_EX

6 推荐序

在流行的编程语言中，Ruby 比较另类，这是因为大多数编程语言的首要着眼点在于为解决特定的问题领域而设计语言，而 Ruby 的首要着眼点在于“人性化”，让程序员充分享受编程的乐趣。由于组织国内 |

列出类/module 的常量	Module#constants
获取常量值	Module#const_get
设置常量值	Module#const_set
删除常量	Module#remove_const
列出类变量	Module#class_variables
获取类变量值	Module#class_variable_get
设置类变量值	Module#class_variable_set
删除类变量	Module#remove_class_variables
定义类方法	Module#define_method
删除类方法	Module#remove_method
解除类方法定义	Module#undef_method
给类方法赋予别名	Module#alias_method
包含模块	Module#include
获取父类	class#superclass

的 Ruby 会议，我曾两次邀请松本行弘来中国。他是一位性格平和、对生活充满热爱的人，在演讲中也一再传递 code for fun 的宗旨，即编程语言不应该是冷冰冰地给机器阅读的执行的指令，而应该让程序员编程的工作过程变成一种充满乐趣和享受的过程。而且，松本先生发明 Ruby 语言也是因为他对创造一种人性化的面向对象脚本语言的热爱。

程序员社区经常拿另外一个主流的面向对象脚本语言 Python 来和 Ruby 做对比。从全球范围来看，Python 的社区更大，应用更广泛，但 Ruby 的语法相对 Python 来说更强大和宽松，给程序员发挥的自由度更大，可以基于 Ruby 创建各个领域的 DSL，比方说 Ruby on Rails 就是一个基于 Ruby 的 Web 快速开发领域的 DSL。

总之，Ruby 语言的这种“人性化”以及给程序员很大编程自由度的气质奠定了整个 Ruby 社区的气质：热爱生活的程序员，所求编程的自由度，带点非主流的极客色彩。也正因为如此，Ruby 和基于 Ruby 的 Rails 得到了硅

谷许许多多创业公司青睐，有名者如 Twitter、Groupon、Hulu、github 等。而这种气质也鲜明地体现在 Rails 框架的创建者 David Heinemeier Hansson 及其所在的 37signals 公司身上。37signals 的 20 多位员工遍全球，每天只上班四天，David Heinemeier Hansson 本人不是一位保时捷车队的职业赛车手。

当然，Ruby 并非只在非主流程序员社区中流行，随着全球 IT 产业进行云计算时代，Ruby 也发挥着越来越大的作用。著名的 SAAS 厂商 salesforce 在 2010 年底以 2.1 亿美元收购了 PAAS 厂商 Heroku，并且在 2011 年 7 月聘请松本行担任 Heroku 首席架构师，开拓 Ruby 在云计算领域的应用。Heroku 本身就是一个完全采用 Ruby 架构的 PAAS 平台，同样支持 Ruby 的 PAAS 厂商还有 EngineYard、VMware 等。随着这些云计算厂商的努力，Ruby 必然在未来得到越来越广泛的应用。我之前阅读了本书的部分章节，这本书实际上是松本行弘从一个编程语言设计者的角度去看待各种各样的流行编程语言，分析它们有哪些特点，以及 Ruby 编程语言是如何取舍的。Ruby 语言的设计本身大量参考了一个更古老而著名的面向对象编程方法的开山之作 Smalltalk，而且从函数式编程语言鼻祖 Lisp“偷师学艺”了不少好东西。程序员社区有个著名的说法：任何现代编程语言都脱胎于 Smalltalk 和 Lisp，都与这两个编程语言有着似曾相识的特性，自 Smalltalk 和 Lisp 诞生以来，编程语言领域可谓大势已定了。因此，集这两种编程语言诸多特点于一身的 Ruby 语言很值得编程爱好者去学习，而看看 Ruby 设计师是怎么设计 Ruby 语言的，则可以让高屋建瓴地理解一些主流的编程语言。

范凯 L^AT_EX

7 中文版序

从年轻的时候开始，我就对编程语言有着极为浓厚的兴趣。比起“使用计算机干什么”这一问题，我总是一门心思想着“如何将自己的意图传达给计算机”。从这个意义上说，我认为自己是个“怪人”。但是，想选择一个能让自己的工作变得轻松的编程语言，想编写一种让人用起来感到快乐的编程语言，一直是我梦寐以求的，这种迫切的心情怕不输于任何人。虽说是有点自卖自夸，但是 Ruby 确实给我带来了“快乐”，这一结果让我感到很满足。

让我感到惊奇的是，有很多人，包括那些没有我这么“怪”的人，都对这种快乐有着共鸣。Ruby 自 1995 年在互联网上公布以来，着实让世界各地的程序员都认识了它，共享着这种快乐，提高了软件开发的生产力。完全出乎我意料的是，世界各地的人，不管是东方还是西方，都极为欣赏 Ruby。在刚开始开发 Ruby 的时候，我想都没有想到过有这样的结果，程序员的感受会超越人种、国籍、文化，有如此之多的共通之处。

现在，为世界各地的程序员所广泛接受的 Ruby，正带来一种新的文化。已经有越来越多的开发人员，在实践中果敢地施行着 Ruby 语言及其社区所追求的“对高生产力的追求”、“富有柔性的软件开发”、“对程序员人性的尊重”、“鼓起勇气挑战新技术”等原则。在 Ruby 以前，这些想法也都很好，却一直实践不起来。我相信，Ruby 的卓越之处，不仅在于语言能力，而且更重要的是引领了这种文化的践行。

本书在解说编程中的技术与原则时，不局限于表面现象，而是努力挖掘其历史根源，提示其本质。虽然很多章节都以 Ruby 为题材，但这些原则对于 Ruby 以外的语言也行之有效。衷心希望大家能够实践本书中所讲述的各项原则，成为一个更好的开发人员。

松本行弘

2011 年 4 月 18 日 L^AT_EX

8 **DONE** 前言 <2016-08-30 二 14:10>

本书的目的不是深入讲解哪种特定的技术，也没有全面讨论我所开发的编程语言 Ruby，而是从全局角度考察了与编程相关的各种技术。读者千万不要以为拿着这本书，就可以按图索骥地解决实际问题了。实际上，最好反它看成是一幅粗略勾勒出了编程世界诸要素之间关系的“世界地图”。

每种技术、思想都有其特定的目的、渊源和发展进步的过程。本书试图换一个角度重新考察各种技术。如果你看过后能够感觉到“啊，原来是这样的呀！”或者“噢，原来这个技术的立足点在这里呀！”那么我就深感心慰了。我的愿望就是这些知识能够激发读者学习新技术的求知欲。

本书的第 2 章到第 14 章，是在《日经 Linux》杂志于 2005 年 5 月到 2009 年 4 月连载的“松本编程模式讲坛”基础上编辑修改而成的。但实际上连载与最开始的设想并不一致，真正涉及“模式”的内容并不多，倒是技术内幕、背景分析等内容占了主流。现在想来，大方向并没有错。

除了连载的内容之外，本书还记录了我对编程问题的新思考和新看法。特别是第 1 章“我为什么开发 Ruby”，针对“为什么是 Ruby”这一点，比其他杂志做了更加深入的解说。另外，在每章的末尾增加了一个小专栏。

对于连载的内容，因为要出成一本书，除修改了明显的错误和不合时代的部分内容之外，我力求每一章都独成一体、内容完整，同时也保留了连载时的风貌。通读全书，读者也许会感到有些话题或讲解是重复的，这一点敬请原谅。

我的本职工作是程序员，不能集中大段时间去写书，不过无论如何最后总算是赶出来了。非常感谢我的家人，她们在这么长时间里宽容着我这个情绪不稳的丈夫和父亲。

稿子写完了，书也出来了，想着总算告一段落了吧，而《日经 Linux》又要开始连载“松本行弘技术剖析”了，恐怕还要继续让家里人劳心。

松本行弘

2009 年 4 月于樱花季节过后的松江