



TURING

图灵程序设计丛书

与全世界程序员分享你的代码

GitHub

入门与实践

[日] 大塚弘记 / 著
支鹏浩 刘 斌 / 译



版本管理 · 分支 · 合并
Fork · Issue · BUG管理
任务管理 · Markdown
代码审查 · 社会化编程 · Jenkins
GitHub Flow · Git Flow · hub命令



中国工信出版集团



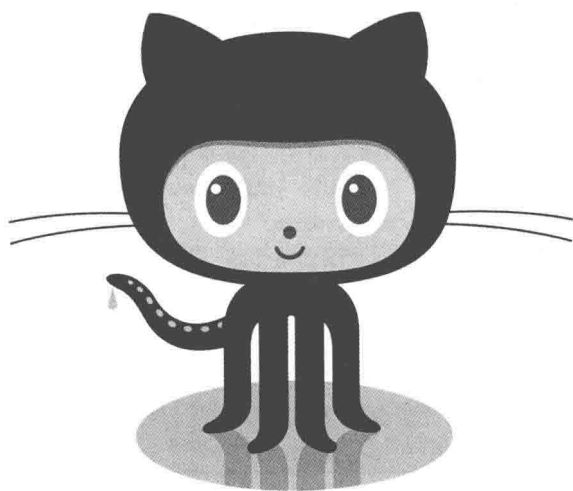
人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

GitHub 入门与实践

[日] 大塚弘记 / 著
支鹏浩 刘 斌 / 译



人民邮电出版社
北 京

图书在版编目(CIP)数据

GitHub入门与实践/(日)大塚弘记著;支鹏浩,刘斌译.--北京:人民邮电出版社,2015.7

(图灵程序设计丛书)

ISBN 978-7-115-39409-5

I. ①G… II. ①大… ②支… ③刘… III. ①软件工具—程序设计 IV. ①TP311.56

中国版本图书馆CIP数据核字(2015)第112943号

内 容 提 要

本书从Git的基本知识和操作方法入手,详细介绍了GitHub的各种功能,GitHub与其他工具或服务的集成,使用GitHub的开发流程以及如何将GitHub引入到企业中。在讲解GitHub的代表功能Pull Request时,本书专门搭建了供各位读者实践的仓库,邀请各位读者进行Pull Request并共同维护。

本书旨在指导各位读者如何在开发现场使用GitHub进行高效开发,适合所有想要使用GitHub进行开发的程序员或团队阅读。

◆ 著 [日]大塚弘记

译 支鹏浩 刘斌

责任编辑 乐 馨

执行编辑 高宇涵

责任印制 杨林杰

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京隆昌伟业印刷有限公司印刷

◆ 开本: 880×1230 1/32

印张: 8.75

字数: 260千字

2015年7月第1版

印数: 1-4000册

2015年7月北京第1次印刷

著作权合同登记号 图字: 01-2015-1263号

定价: 39.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第0021号

● 译者序

“开源”一词在我国 IT 界已经出现了不少年头，但“社会化编程”想必没有多少人接触过。于是在阅读正文之前，容我越俎代庖替作者问一个问题：各位在狭小的空间里呆上一段时间之后，再出门时是否有一种豁然开朗的感觉？相信很多人的答案都是肯定的。对于对日外包出身的我来说，“社会化编程”就给了我这种感觉。或许外包行业在 IT 界只是极端个例，但“让全世界码农看自己的代码”这种事，很多人恐怕想都不敢想吧。

GitHub 正是这样一个平台，我们在这里可以与全世界的开源开发者交流代码或心得。如果您对某款开源软件的源代码感兴趣，如果您想为中意的软件出一份力，如果您自己编写了小程序却苦苦找不到人指点，如果您想跟慕名已久的 IT 界明星（俗称“大神”）聊上几句，那么 GitHub 欢迎您。

GitHub 的纯英文界面或许会令您望而却步，不过不用担心，本书秉承了日系技术书刊一贯的“手把手教学”风格，作者用亲切的语言，简明扼要的介绍，配以生动详实的示例为我们一步步讲解 GitHub 的使用方法，带我们在实践中学习 GitHub。值得一提的是，本书配有一个供各位实践的网站，请感兴趣的读者务必一试。俗话说“读万卷书不如行万里路”，跟着作者一边实践一边阅读本书，相信各位会对这句话有一个更深刻的体会。

有些读者可能要问了，代码是企业的财产，不能随便发到网上给别人看，那 GitHub 对工作又有什么意义呢？这一点作者自然考虑到了。GitHub 面向社会化编程，我们所生活的是一个社会，我们工作的企业同样是一个小社会，虽然不能强行导入“社会化编程”，但其管理模式仍然值得借鉴。所以如果您是企业的决策者，那么请在本书后半跟随作者一起探讨企业导入社会化编程的利弊，说不定能为您所在的企业带来新的利益。

《GitHub 实战入门》是国内比较少见的对 GitHub 及社会化编程进行系统介绍的书籍。以往我们对于这方面知识，只能通过网络上零零散散

的博客或技术文档进行片面了解，难以把握其全貌。各位读完这本书后相信能得到不少帮助。

最后，对另一位帮忙搭建本书相关网站的译者以及图灵文化的各位编辑致以衷心的感谢，正是有了各位的共同努力，本书才得以出版。同时感谢正在阅读本书的您，有了您的支持，本书才能发挥其价值。

支鹏浩

2015 年 4 月 于北京

● 序言

当今世界有众多开发者在使用 GitHub 进行开发。本书旨在指导各位读者在开发现场如何使用 GitHub 进行高效开发。因此，书中除针对 GitHub 进行讲解外，也涉及了开发流程及相关辅助工具的解说。

您在开发现场有没有遇到过以下几件事？

- 代码审查不到位，审查效率低下
- 只有编程者本人能看懂的代码、可靠性不高的代码直接被部署至正式环境中
- 因键入错误、理解错误而造成的低级代码错误导致 BUG 频繁出现
- 没有机会和其他人互相交流代码，共享知识，相互学习、指正、改善
- 没有一个简单高效、能在一天之内添加多个功能的开发流程

GitHub 为我们提供了解决这些问题的机会和功能，而本书则凝练了各种运用 GitHub 的诀窍。

笔者曾为多家企业引入 GitHub，改善其开发流程。本书总结了这些经验，相信能为改善您的开发现场提供一些帮助。

●……谢辞

本书在编撰过程中得到了多方的大力支持。特此鸣谢 @yamanetoshi、增田贵士 (@masutaka)、bakorer、山科佑贵、寺田涉、Tatsuma Murase、杉野康弘、泽义和（排名不分先后）。

另外，长期以来，技术评论社的池田大树为本书的编辑与整理尽心尽力，在此由衷地表示感谢。

2014 年 2 月 大塚弘记

● 本书结构

本书由 10 章及 2 个附录构成。

第 1 章：欢迎来到 GitHub 的世界

讲解 GitHub 是什么，以及有哪些革新之处。在开源软件的世界中，GitHub 为开发者带来了革命性的社会化编程概念。在这里我们将会接触这一概念，并对其带来的优势与功能进行讲解。

第 2 章：Git 的导入

要使用 GitHub，离不开 Git 这一版本管理系统。本章将深入介绍关于 Git 的知识，加深各位对 Git 的理解，同时说明实际操作的相关流程。

第 3 章：使用 GitHub 的前期准备

使用 GitHub 需要开设账户（免费），因此我们将按照顺序为您讲解正式使用前需要进行的一系列设置。

另外，本章还会讲解包括操作示例在内的，实际在 GitHub 上创建仓库并发布代码的相关流程。

第 4 章：通过实际操作学习 Git

在实际操作中学习使用 GitHub 时所必需掌握的 Git 的基本知识和操作方法。

从最基本操作到多人开发时所需的复杂操作，读者都可以随着本章的讲解简单实践一番。

第 5 章：详细解说 GitHub 的功能

本章我们将以图配文，对 GitHub 的功能逐一进行讲解，同时还会详细解说其作为源代码查看器的功能，带您领略方便快捷的 UI。

建议正在使用 GitHub 的开发者也读一读本章，您或许会发现一些将来能用到的小技巧。

第 6 章：尝试 Pull Request

Pull Request 是 GitHub 的代表功能，本章我们将带您亲自动手体会。请务必参考本书内容试着进行一次 Pull Request。

第 7 章：接收 Pull Request

站在仓库维护方的角度，教您在接到 Pull Request 之后应该如何考虑，如何判断，以及该进行哪些操作。

第 8 章：与 GitHub 相互协作的工具及服务

前半部分为您讲解通过 CLI 对 GitHub 进行操作时所需的 hub 命令。另外，在持续集成环境方面，将讲解可与 GitHub 结合使用的 Travis CI 及 Jenkins 的构建及设定方法。

除此之外，本章还会介绍一些能够与 GitHub 共同使用的服务。

第 9 章：使用 GitHub 的开发流程

详细讲解以 GitHub 为中心进行开发的 GitHub Flow、Git Flow 两个开发流程。从两者共通的团队开发心得到各自开发流程的特征，都可以通过本章的讲解实际动手体会。

第 10 章：将 GitHub 应用到企业

总结在企业中采用 GitHub 时需要考虑的问题及一些有用的信息。安全保障、故障信息、事前需要考虑的问题、GitHub Enterprise 的讨论等，这些实际引入 GitHub 时需要考虑或者了解的知识将在本章中进行讲解。

附录 A：辅助 GitHub 的 GUI 客户端

团队中并不是每个人都对 CLI 得心应手。因此，我们为读者总结了辅助 GitHub 的 GUI 客户端的相关知识。

附录 B：通过 Gist 轻松实现代码共享

Gist 能帮助开发者轻松与其他人共享简单的代码示例或日志，我们将在这部分对 Gist 进行讲解。利用 Gist 可以轻松管理日常的小代码片段。

本书内容以敝社《WEB+DB PRESS》Vol.69 的特辑《详解 GitHub——使用 Pull Request 打造高效率的软件开发》^①为基础，进行大篇幅扩展与修正后作为图书出版。

本书的操作示例是在以下环境中进行的。

- OS X 10.9.1
- git 1.8.5.2

部分 Windows 相关解说中使用了 Windows 8。另外，GitHub 相关解说皆以 2014 年 2 月时的版本为基准。

由于环境和时期不同，操作顺序、页面、运行结果可能会存在差异。

本书中出现的示例仓库，现阶段主要由译者及尝试 Pull Request 的各位读者进行维护。但是在本书出版后，随着时间推移，可能会发生反应变慢甚至没有反应的情况。烦请参照第 7 章的内容以及关于示例仓库的讲解，一同努力维护。

对于您应用本书内容所产生的后果，本书作者、软件开发方及供应方、技术评论社、人民邮电出版社及译者概不负责，特在此声明。

本书中提及的公司名、制品名，皆是各公司实际使用的注册商标或商标。在正文中并未添加™、©、® 标志。

关于本书的补充信息与勘误等，请参考以下网址。

<http://www.ituring.com.cn/book/1581>

① 详解 GitHub——Pull Request が織りなす効率的のソフトウェア開発，WEB+DB PRESS vol.69，技术评论社。——译者注

● 目录

	第1章 欢迎来到GitHub的世界	1
1.1	什么是 GitHub	2
	GitHub 公司与 octocat	2
	并不只是 Git 仓库的托管服务	3
	GitHub 的使用情况	3
	Column 专栏：GitHub 与 Git 的区别	4
1.2	使用 GitHub 会带来哪些变化	4
	协作形式变化	4
	在开发者之间引发化学反应的 Pull Request	5
	对特定用户进行评论	6
	GitHub Flavored Markdown	7
	Column 专栏：还可以这样写 !!	7
	能看到更多其他团队的软件	7
	与开源软件相同的开发模式	8
1.3	社会化编程	9
1.4	为什么需要社会化编程	10
	不要闭目塞听，要接触不同的文化	10
	会写代码的程序员更受青睐	11
	GitHub 最大的特征是“面向人”	11
1.5	GitHub 提供的主要功能	12
	Git 仓库	12
	Organization	12
	Issue	13
	Wiki	13
	Pull Request	13
	Column 专栏：GitHub 上受到瞩目的软件	14
1.6	小结	14

参考资料	14
------	----

第2章 Git的导入 17

2.1 诞生背景	18
----------	----

2.2 什么是版本管理	18
-------------	----

集中型与分散型	19
---------	----

集中型	19
-----	----

分散型	19
-----	----

集中型与分散型哪个更好	20
-------------	----

2.3 安装	21
--------	----

Mac 与 Linux	21
-------------	----

Windows	21
---------	----

组件的选择	22
-------	----

设置环境变量	22
--------	----

换行符的处理	23
--------	----

Git Bash	23
----------	----

本书所用的环境	24
---------	----

2.4 初始设置	24
----------	----

设置姓名和邮箱地址	24
-----------	----

提高命令输出的可读性	25
------------	----

2.5 小结	25
--------	----

第3章 使用GitHub的前期准备 27

3.1 使用前的准备	28
------------	----

创建账户	28
------	----

设置头像	29
------	----

设置 SSH Key	29
------------	----

添加公开密钥	30
--------	----

使用社区功能	31
--------	----

3.2 实际动手使用	31
创建仓库	31
Repository name	32
Description	32
Public、Private	32
Initialize this repository with a README	32
Add .gitignore	33
Add a license	33
连接仓库	33
README.md	33
GitHub Flavored Markdown	34
公开代码	34
clone 已有仓库	34
编写代码	35
提交	36
Column 专栏：公开时的许可协议	37
进行 push	37
3.3 小结	38

第4章 通过实际操作学习 Git 39

4.1 基本操作	40
git init——初始化仓库	40
git status——查看仓库的状态	40
git add——向暂存区中添加文件	41
git commit——保存仓库的历史记录	42
记述一行提交信息	42
记述详细提交信息	42
中止提交	43
查看提交后的状态	43
git log——查看提交日志	43
只显示提交信息的第一行	44
只显示指定目录、文件的日志	44
显示文件的改动	45

git diff——查看更改前后的差别	45
查看工作树和暂存区的差别	45
查看工作树和最新提交的差别	46
4.2 分支的操作	47
git branch——显示分支一览表	48
git checkout -b——创建、切换分支	48
切换到 feature-A 分支并进行提交	48
切换到 master 分支	49
切换回上一个分支	50
特性分支	50
主干分支	51
git merge——合并分支	51
git log --graph——以图表形式查看分支	52
4.3 更改提交的操作	53
git reset——回溯历史版本	53
回溯到创建 feature-A 分支前	53
创建 fix-B 分支	54
推进至 feature-A 分支合并后的状态	55
消除冲突	56
查看冲突部分并将其解决	57
提交解决后的结果	57
git commit --amend——修改提交信息	58
git rebase -i——压缩历史	59
创建 feature-C 分支	59
修正拼写错误	60
更改历史	61
合并至 master 分支	63
4.4 推送至远程仓库	63
git remote add——添加远程仓库	64
git push——推送至远程仓库	64
推送至 master 分支	64
推送至 master 以外的分支	65
4.5 从远程仓库获取	65
git clone——获取远程仓库	65

获取远程仓库	65
获取远程的 feature-D 分支	66
向本地的 feature-D 分支提交更改	67
推送 feature-D 分支	67
git pull——获取最新的远程仓库分支	67
4.6 帮助大家深入理解 Git 的资料	68
Pro Git	68
LearnGitBranching	69
tryGit	69
4.7 小结	70

第5章 详细解说GitHub的功能 71

5.1 键盘快捷键	72
5.2 工具栏	73
关于 UI	73
1 LOGO	73
2 Notifications	73
3 搜索窗口	73
4 Explore	73
5 Gist	74
6 Blog	74
7 Help	74
8 头像、用户名	74
9 Create a new...	74
10 Account settings	75
11 Sign out	75
5.3 控制面板	75
关于 UI	75
1 News Feed	76
2 Pull Requests	76
3 Issues	76
4 Stars	76
5 Broadcast	76

6	Repositories you contribute to	76
7	Your Repositories	76
5.4	个人信息	77
	关于 UI	77
1	用户信息	77
2	Popular Repositories	78
3	Repositories contributed to	78
4	Public contributions	78
5	Contribution Activity	78
6	Repositories	78
7	Public Activity	79
5.5	仓库	80
	关于 UI	80
1	用户名 (组织名) / 仓库名	80
2	Watch/Star/Fork	80
3	Code	81
4	Issue	81
5	Pull Requests	81
6	Wiki	82
7	Pulse	82
8	Graphs	82
9	Network	82
10	Settings	82
11	SSH clone URL	82
12	Clone in Desktop	82
13	Download ZIP	83
a	commits	83
b	branches	83
c	releases	83
d	contributors	83
e	Compare & review	83
f	branch	83
g	path	84
h	Fork this project and Create a new file	84
i	files	84
	文件的相关操作	84
	Column 专栏 : 通过部分名称搜索文件	85

查看差别	85
查看分支间的差别	85
查看与几天前的差别	86
查看与指定日期之间的差别	87
5.6 Issue	87
简洁且表现力丰富的描述方法	88
语法高亮	89
添加图片	90
添加标签以便整理	90
添加里程碑以便管理	91
Column 专栏：了解贡献时的规则！	92
Tasklist 语法	92
通过提交信息操作 Issue	93
在相关 Issue 中显示提交	93
Close Issue	93
将特定的 Issue 转换为 Pull Request	94
5.7 Pull Request	94
Column 专栏：获取 diff 格式与 patch 格式的文件	96
Conversation	96
Column 专栏：引用评论	96
Commits	97
Column 专栏：在评论中应用表情	98
Files Changed	98
5.8 Wiki	99
Pages	100
History	101
Column 专栏：在 Wiki 中显示侧边栏	101
5.9 Pulse	102
active pull requests	103
active issue	103
commits	104
Releases published	104
Unresolved Conversations	104

5.10 Graphs	105
Contributors	105
Commit Activity	106
Code Frequency	106
Punchcard	108
5.11 Network	108
5.12 Settings	109
Options	109
❶ Settings	109
❷ Features	110
❸ GitHub Pages	111
❹ Danger Zone	111
Collaborators	111
Webhooks & Services	112
Deploy Keys	112
5.13 Notifications	112
5.14 其他功能	114
GitHub Pages	114
GitHub Jobs	114
GitHub Enterprise	114
GitHub API	115
5.15 小结	115
Column 专栏：在 Mac 的通知中心查看 GitHub 的 Notifications	115

第6章 尝试 Pull Request 117

6.1 Pull Request 的概要	118
什么是 Pull Request	118
Pull Request 的流程	118
6.2 发送 Pull Request 前的准备	119
查看要修正的源代码	120