

## 如何高效利用GitHub

**正** 是GITHUB，让社会化编程成为现实。本文尝试谈谈GIT HUB的文化、技巧与影响。

- [Q1: GitHub是什么](#)
- [Q2: GitHub风格](#)
- [Q3: 在GitHub，如何跟牛人学习](#)
- [Q4: 享受纯粹的写作与演讲](#)
- [Q5: 代码帮你找工作](#)
- [Q6: GitHub还在影响一些什么](#)
- [Q7: 除了GitHub，还可以选择什么?](#)

### Q1: GitHub是什么

**A1: 一家公司**



位于旧金山，由[Chris Wanstrath](#), [PJ Hyett](#) 与[Tom Preston-Werner](#)三位开发者在2008年4月创办。迄今拥有59名全职员工，主要提供基于git的版本托管服务。

在此之前，它是由Tom与Chris在本地程序员聚会中，开始的一个用于托管git的项目。正如每个伟大的传奇都开始于一场冒险，Tom在这篇文章[我如何辞掉微软30万年薪邀约，创办GitHub](#)中谈到：

*当我老去，回顾一生，我想说，“哇，那是一场冒险”；而不是，“哇，我真的很安稳。”*

另一位创始人Chris也详细描述了GitHub初创的前因后果，他说道：

*Do whatever you want.*

于是，在2008年4月10号这一天，GitHub正式成立。

目前看来，GitHub这场冒险已经胜出。根据来自[维基百科关于GitHub的描述](#)，我们可以形象地看出GitHub的增长速度：



今天，GitHub已是：

- 一个拥有143万开发者的社区。其中不乏Linux发明者Torvalds这样的顶级黑客，以及Rails创始人DHH这样的年轻极客。
- 这个星球上最流行的开源托管服务。目前已托管431万git项目，不仅越来越多知名开源项目迁入GitHub，比如Ruby on Rails、jQuery、Ruby、Erlang/OTP；近三年流行的开源库往往在GitHub首发，例如：[BootStrap](#)、[Node.js](#)、[CoffeScript](#)等。
- alexa全球排名414的网站。

## Q2: GitHub风格

### A2: GitHub只是GitHub

强调敏捷开发与快速原型，而又的确成功的创业团队，常具备一个重要气质：有自己的文化风格。如GitHub，又如[37signals](#)。通过他们的快速开发，向用户证明了团队在技术上的能力，并且时常有惊喜。同时，通过强调特立独行的文化，将对半衰期过短的产品族群的信任转为对GitHub团队的信任。

[Gravatars](#)的创始人（对，就是互联网最流行的头像托管系统）、[Jekyll](#)（对，它就是我近几年用的博客系统）作者、GitHub创始人，现任CTO Tom在[GitHub第一年学到的10大教训、创业学校演讲](#)中谈到GitHub文化的方方面面。我尝试将这种风格总结为以下要点：

- 专注创作，高创意
- 运营良好与较高的内外满意度
- 高利润，较低的融资额或零融资

创业公司多半死在钱上，就让我们先从钱谈起：

#### 高利润，较低的融资额或者零融资

类似于GitHub这样的公司，拿到风险投资很难吗？恰恰相反，创始人[PJ Hyett](#)在Hacker News的[一篇评论](#)中提到，自从GitHub创办以来，已与几十个VC沟通过。但是，直到今天，GitHub的融资额还是为零，并引以为豪。让我们看看GitHub官网的自我介绍：

Box of Fact™	
Home Base	San Francisco
Company Founded	2008
Hubbernauts Employed	59
Delicious Beers On Tap	4
VC Funding	\$0.00

## 运营良好与较高的内外满意度

在Quora上有人问道，[GitHub](#)是否寻找被收购？，还是[PJ Hyett](#)，他的回答是：No。

GitHub从一开始就运营良好，员工拥有较高满意度，看看这些不太一样的做法：

- 每一位GitHub公司的新员工，官方博客将发表文章欢迎。
- 在GitHub内部，没有经理，需求内容与优先级由项目组自行决策。
- 选择自己的工作时间、工作地点。
- 员工来自开源社区。
- 能开源的尽可能开源。

富有激情、创意的员工使得GitHub得到了社区的广泛认同，从而拥有极高的客户满意度，

并从创业一开始就盈利。一份[早期的调查](#)表明，GitHub很快成为Git托管首选。

## 专注创作，高创意

GitHub 59名全职员工仅有29名员工在本地工作！不仅仅是工作地点的安排富有创意，GitHub员工[Holman](#)，详细介绍了GitHub的工作方式：

- 时间并不能说明什么
- 异步工作方式
- 创造力很重要

## Q3:在GitHub，如何跟牛人学习

## A3:在学习区刻意练习

### 追随牛人，与他们一起修行

修行之道：

关注大师的言行，  
跟随大师的举动，  
和大师一并修行，  
领会大师的意境，  
成为真正的大师。

正如这首[禅诗](#)所言，与其在墙内仰望牛人，不如直接在GitHub：

- [watch](#)、[fork](#)牛人们
- 对他们的项目提交[pull request](#)
- 主动给牛人们的项目写[wiki](#)或提交测试用例，或者问题
- 还可以帮他们翻译中文

GitHub本身建构在git之上，git成为勾搭大师们的必要工具，以下读物成为首选：

- [git大白话入门](#)，木有高深内容
- [为什么git胜过X...](#)

如果希望进一步深入，可以阅读已有中文翻译版的材料：

- [progit](#)：GitHub公司传道士[schacon](#)所作，已翻译成多国语言，当然，有中文版。
- [Git Magic](#)：已有志愿者翻译[中文版](#)。

同样，如果希望了解更多GitHub自身的知识，GitHub官方文档值得推荐：

- [The GitHub Hep](#)

## 牛人在哪里？

- GitHub上的代码库本身：尤其是：[Explore](#)、[热门关注信息库](#)两个栏目
- GitHub官方推荐：[GitHub自身的官方博客](#)与GitHub员工们的个人博客推荐的项目与开发者
- 各类社交媒体上提到的的GitHub库：尤其是[Hacker News](#)上提到的GitHub库。

关于学习的心理学研究，常常会谈到术语：**元认知、元学习、元知识**。是的，关于认知的认知、关于学习的学习、关于知识的知识，你对这些信息的偏好与熟练掌握，会让你在学习一门新东西时更加轻车熟路。**对一手信息进行回溯**，比如作者、创始人、最初文献出处，总是会让你更容易理解知识。

## 在学习区刻意练习：借助GitStats进行项目统计

在[如何学习一门新的编程语言？](#)——在学习区刻意练习中，我已谈过：

*学习编程最好的方式是在学习区刻意练习。*

如何进行自我监督？

借助于[GitStats](#)，我们能很好地统计自己的每个项目的工作量，从而看到工作进展。

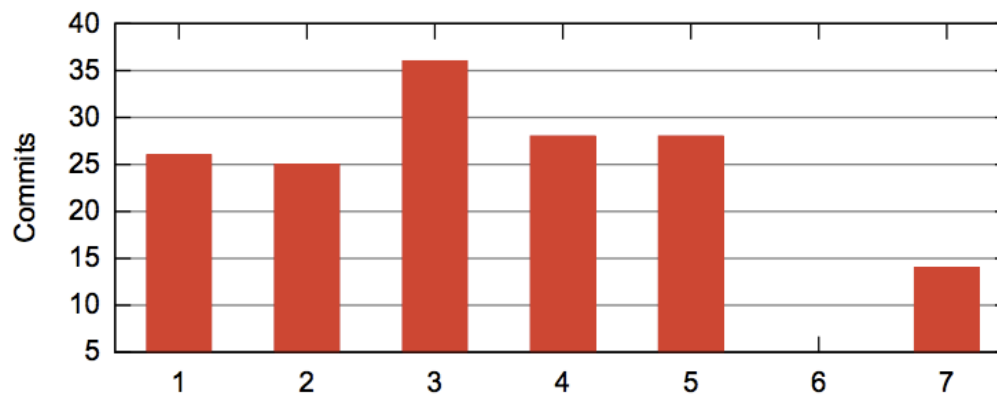
用法如下，

```
#复制GitStats项目到本地
cd ~/dev
git clone git://github.com/trybeee/GitStats.git
python ~/dev/gitstats/git-stats /youproject public
```

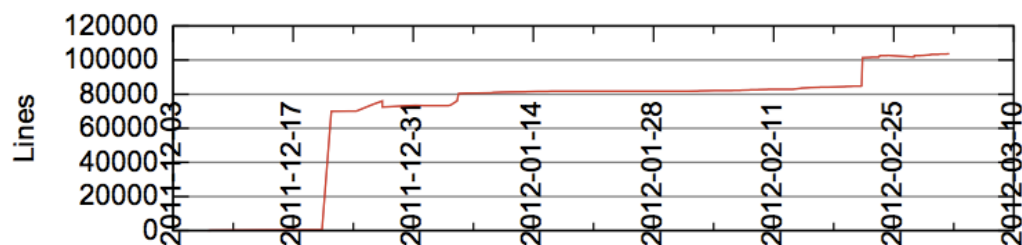
以下为生成结果示范：

每周代码提交次数：





每天代码提交行数:



如果Fork别人的项目或者多人合作项目，最好每人都拥有一个独立分支，然后由项目维护人合并。如何建立自己的分支？

```
# 分支的创建和合并
# git branch yourbranch
# git checkout yourbranch    切换到yourbranch

# 开发yourbranch分支，然后开发之后与master分支合并
```

```
# git checkout master
# git merge yourbranch
# git branch -d yourbranch    合并完后删除本地分支
```

如何将牛人的远程分支更新到自己的本地分支？

```
# 查看当前项目下远程
# git remote
# 增加新的分支链接，例如
git remote add niuren giturl...
# 获取牛人的远程更新
git fetch niuren
# 将牛人的远程更新合并到本地分支
git merge niuren/master
```

## 生产力小技巧

### codeshelfer: 给git库做标签

观察的项目如果多了，怎么管理？用[codeshelfer](#)，安装扩展之后，可以对GitHub项目做标签。

### gollum: 利用git与github做wiki

[gollum](#)是一个基于git的轻型wiki系统。

### GitHubwatcher: 监测重点项目

[GitHubwatcher](#)适用于通知不频繁的情景。

## GitHub官方资源

GitHub官方列出了一些有用的脚本与书签。

### 社区驱动的安装与配置文件

GitHub中各类配置文件层出不穷，一些常用的：

- [osh-my-zsh](#)：将终端从bash改为zsh之后，可考虑安装社区驱动的zsh配置文件，含有多  
个插件。可参考旧文[zsh与oh-my-zsh](#)
- [gitignore](#)：GitHub官方出品
- [yourchili](#)：服务器各类安装shell，比如安装nginx等。

## Q4: 享受纯粹的写作与演讲

### A4：回归创作的初始

#### 写作

早在2008年，就有技术图书作者[通过Git来写作](#)，以下是示范：

- [Node.js初学者教材](#)，中文版[在这里](#)。
- [backbone基础](#)
- [Sinatra教程](#)

你能想到的技术前沿话题，大多能在GitHub找到相应的培训材料或者开源图书。

个人写作照样适用。在前文理想的写作环境：[Git+GitHub+Markdown+Jekyll](#)，我已经格外赞美过这些美好事物了。

暖色调的灯光，足够宽度的工作台，听着清脆的键盘声音，基于Git、GitHub、Markdown与Jekyll来写作，不担心写废与排版，只关注最纯粹的写作，是一种享受。我有时候会想，如果Git、Github、Markdown、Jekyll，再加上Yaml、Json的作者，让这些作者们重新来设计今天互联网基础架构偏文本的部分，会诞生一些什么？

## 个人博客

借助于[Jekyllbootstrap](#)，可以在Github上快速搭建一个基于jekyll的博客系统。

除了这个简单易行的办法之外，还存在一些其他方法，例如：

- Jekyll：参考[告别wordpress，拥抱jekyll](#)
- Octopress：参考[Ruby开源项目介绍\(1\): octopress——像黑客一样写博客](#)
- GitHub Pages：参考[GitHub Pages](#)

## 演讲

借助于GitHub，可以享受更纯粹、更酷的演讲。GitHub 2011年收购Ordered List之后，从此可以通过[speakerdeck](#)更好的分享ppt文档。

我们还可以：

- 使用GitHub著名传教士、Progit作者Scott Chacon开发的[showoff](#)
- 来自开源社区的其他演讲库[impress.js](#)

## Q5: 代码帮你找工作

### A5: GitHub简历很诚实

NumEricR（非GitHub工作人员）基于GitHub Pages功能做了一个简历生成器，使用极其简单，登陆网站[GitHub简历生成器](#)，填入你的GitHub网站用户名即可。

fredwu是Ruby中文社区活跃份子，他的开源项目[angel\\_nest](#)，一个天使投资与创业者对接的网站，适合Ruby初学者升级为Ruby中级开发者时学习，也在Hacker News上被[热烈讨论](#)过，让我们来看看他的简历：

<http://resume.GitHub.com/?fredwu>

正是因为GitHub上的代码无法造假，也容易通过你关注的项目来了解知识面的宽度与深度。现在越来越多知名公司活跃在GitHub，发布开源库并招募各类人才，例如：[Facebook](#)、[Twitter](#)、[Yahoo](#) ...

开始有了第三方网站提供基于GitHub的人才招聘服务，例如：

- [GitHire](#):通过它，可以找出你所在地区的程序员。
- [Gitalytics.com](#): 通过它，能评估某位程序员在GitHub、LinkedIn、StackOverflow、hackernews等多个网站的影响力。

## Q6: GitHub还在影响一些什么

### A6: 让计算机增强人类智慧

很多年前，在某个名声显赫的学府中，两位先后拿过图灵奖的牛人有一段对话：

- 牛人A：我们要给机器赋予智慧，让他们有自我意识！
- 牛人B：你要给机器做那么多好事？那你打算给人类做些什么呢？

这段对话来自《失控》。牛人A是[明斯基](#)，他最喜欢将人类看做有血肉的机器，他的框架理论成为认知心理学、人工智能入门基础。牛人B则是[恩格尔巴特](#)。当明斯基1961年发表他著名的文章[人工智能走向时](#)，恩格尔巴特还籍籍无名。直到次年，恩格尔巴特发表宏文：[人类智力的增强：一种概念框架](#)。提出不同于明斯基的另一条增强人类智力的道路：不要尝试发明自动打字的机器，而是尝试发明鼠标，并且他真的发明鼠标成功了！

从近些年的发展来看，仍然是明斯基占上风，但是，三十年河东，三十年河西，明斯基的人工智能方向又有多少年没有大突破了？相反，来自恩格尔巴特的群件、集体智慧等思想，逐步成为步入Web2.0时代之后的共识。无关对错，可以说，恩格尔巴特为增强人类智力，提供了可行的框架。与其去发明聪明的、昂贵的、功能一体化的智能机器人，还不如发明类似于鼠标这样笨笨的、廉价的、功能单一的人类智慧服务单件。明斯基的机器人很容易陷入死胡同，没有上升到哲学的高度。现在慢慢又回到恩格尔巴特这个方向来了。比如现在IBM开始[宣传的认知计算](#)。

从git与GitHub设计与解决的问题本质来看，明显加速了代码生产流程，促进了卓越智力产品的诞生。这就是一种典型的web2.0对智力生产流程的改良与人类智慧的增强。同样，某种意义上，小说写作网站也起到类似作用。但是，学术界尤其是社会科学类的智力产品生

产似乎还停留在一个古老阶段。在开源领域，好想法层出不穷，极客影响极客，最终产生的是酷玩意。这些酷玩意抛弃浮华，直奔问题本质。那么，[有没有科学界的GitHub](#)？

类似问题层出不穷，以下为其他领域产品不完全名单。

## 学术研究

- 除了较早的[arXiv](#)、[PLoS](#)之外，较有气象的可以推荐[mendeley](#)、[开放期刊目录](#)

## 数据

- [buzzdata](#): 数据分享更容易

## 科学计算

- [opani](#): 雏形中，支持R、Python等多种。

## 教育

- [OpenStudy](#): 一个社会性学习网络，通过互助来更好地学习，主题涉及到计算机、数学、写作等。
- [openhatch](#): 通过练习、任务等帮助新手更好地进入开源社区

## Q7:除了GitHub，还可以选择什么？

## A7: nil

因为进化的需要，多数裸猿存在选择强迫症：哪种程序语言更好？哪个web开发框架更好？当然，最令宅男技术男们羡慕的问题是，高白瘦御姐还是青春小萝莉好？:D

除了GitHub之外，

- 中国山寨品是不是更好？（为什么不写他们名字，你懂的，山寨品总是善于争论谁是第一个山寨的，各自的排名先后:D）
- 免费的BitBucket是不是更适合Python程序员？
- 作为一名折腾族，我不自己搭建一个gitlabhq，是不是对不起自己？

我们可以理解，正是因为无数条分岔路口，让人类不再受制于某种基因、特定疾病、独裁家族，从而拥有无限的可能。但是，这种选择强迫症与远古时代可怜的信息量相比较，

- 今天这个大数据时代，它还会有助于人类作为族群的整体进化与作为个体的幸福吗？
- 今天一位一线城市30岁大学毕业生经历的选择与孔子整个一生经历的选择，纯论数量，谁多谁少？

生命如此短暂，为什么总要将青春浪费在不断的选择之中呢？罚你，回头阅读心理学家施瓦茨（Barry Schwartz）的TED演讲：[选择之困惑——为何多即是少](#)，1百遍啊1百遍。请记住施瓦茨的演讲要点：

- 更多的选择不代表更多的自由；
- 更多的选择导致决策的延迟和降低的满意感；
- 快乐之秘诀，在于降低自己的期望值。



最后，让我再抒情一把吧，

美好的事物总是离不开被墙的命运，让我们静静地期待那一天的来临... 也让我们在各自行业的努力，让下一代、下一代、下一代...（希望 $N \leq 1$ ，如果 $N \geq 4$ ，我做鬼也放不过你们！）不再拥有这一天。

## 相关参考

- 理想的写作环境：Git+GitHub+Markdown+Jekyll
- 如何提高创作型任务的效率？
- Ruby开源项目介绍(1)：Octopress——像黑客一样写博客
- Git与GitHub入门资料
- 告别WordPress，拥抱Jekyll

本作品采用知识共享署名-非商业性使用-禁止演绎 3.0 Unported许可协议进行许可。

分享到：

[更多](#)

04 MARCH 2012

106 条评论。 · 5 条反应

★ 73



留言...

最好的 ▾ 社区

分享 ▾ ▾

[shirayuan](#) 44 小时前

**pniree yuen** · 11个月前

末尾的抒情是亮点。  
钦此。

4 ^ ^ · 回复 · 分享

**Bbf266** · 1年前

Nice

2 ^ ^ · 回复 · 分享

**haiweili** · 4个月前

I like this .

1 ^ ^ · 回复 · 分享

**袁源** · 5个月前

收益很大!! 谢谢!!

更多的选择不代表更多的自由; 更多的选择导致决策的延迟和降低的满意感; 快乐之秘诀, 在于降低自己的期望值。

1 ^ ^ · 回复 · 分享

**laughing** · 8个月前

好文章, 记得半年前看过一次, 今天再次碰到, 又从头到尾读了一遍, 还是有收获。

1 ^ ^ · 回复 · 分享

**YC** · 9个月前

受教! 博主很有才!

1 ^ ^ · 回复 · 分享

**Guest** · 11个月前

赞! 学习了, 增加了对GitHub的了解。

1 ^ ^ · 回复 · 分享

**noursen** · 1年前



reaction · 1天前  
大赞！好文！！

1 ^ ▾ · 回复 · 分享



Luser News · 1年前  
信息量很大.

1 ^ ▾ · 回复 · 分享



kimigao · 1年前  
very good!

1 ^ ▾ · 回复 · 分享



运韬 李 · 6天前  
信息量真大，楼主有才

0 ^ ▾ · 回复 · 分享



weslee · 12天前  
更多的选择不代表更多的自由；  
更多的选择导致决策的延迟和降低的满意感；  
快乐之秘诀，在于降低自己的期望值。  
喜欢！感谢分享.

0 ^ ▾ · 回复 · 分享



暴君@提不起劲 · 23天前  
为什么git胜过X...这个链接失效了

0 ^ ▾ · 回复 · 分享



futeng · 23天前  
超大信息量，感谢作者幽默又详实的表述！

0 ^ ▾ · 回复 · 分享



**pengtao** · 25天前

博主真有才，写出这么优秀的文章，实在是我们it界之福，希望能不断坚持下去。

0 ^ | v · 回复 · 分享 ›



**biby** · 1个月前

恩，听不错，mark下

0 ^ | v · 回复 · 分享 ›



**酷剑** · 1个月前

看完这篇文章，感觉github更有爱了。

0 ^ | v · 回复 · 分享 ›

**pipi** · 1个月前

nice，不错

0 ^ | v · 回复 · 分享 ›



**joni** · 1个月前

难得一见的有水准的文章。

0 ^ | v · 回复 · 分享 ›

**Xy2401** · 1个月前

看完了，很不错的，最近想学一下git，使用github托管

0 ^ | v · 回复 · 分享 ›

**Leo Wdls** · 1个月前

I like it. it is succinct and have much message

0 ^ | v · 回复 · 分享 ›

**smalltalk voice** · 2个月前

好的太震撼了，来回翻看了好几遍，这样的好文章应该包含付费连接才行，这样才能对的起你的付出和努力！

0 ^ ▾ · 回复 · 分享 ›



**young0512** · 2个月前

好文章，收获很多，接下来还需要慢慢消化。

0 ^ ▾ · 回复 · 分享 ›

**Guest** · 2个月前

学习了，收获很多，有待于慢慢消化。

0 ^ ▾ · 回复 · 分享 ›



**changhu huang** · 2个月前

信息量很大啊，受益了。膜拜大神

0 ^ ▾ · 回复 · 分享 ›

**tenacy** · 2个月前

GITHUB要感谢开复老师...

0 ^ ▾ · 回复 · 分享 ›

**Lions88** · 2个月前

怒赞

0 ^ ▾ · 回复 · 分享 ›



**Yunlong Zhong** · 2个月前

CoffeeScript 少打一个 e

0 ^ ▾ · 回复 · 分享 ›

**jashon wang** · 2个月前

好 感受颇丰

0 ^ ▾ · 回复 · 分享 ›



**Ming Leung** · 3个月前

它但十亿了 然而一个！



与1守从灯」，念以「」！

0 ^ ▾ · 回复 · 分享 ›

---

**Maoran Bian** · 3个月前

尤喜欢最后的抒情哈哈

0 ^ ▾ · 回复 · 分享 ›

---

**Ork** · 3个月前

感到羞愧啊，天天泡网上，这么多好东西都未曾接触和了解过，好文！赞！

0 ^ ▾ · 回复 · 分享 ›

---



**young40** · 3个月前

good

0 ^ ▾ · 回复 · 分享 ›

---



**Richard Lea** · 3个月前

我也想说，为什么这网站也被墙了~~~

0 ^ ▾ · 回复 · 分享 ›

---

**John Hoo** · 3个月前

赞！~~

0 ^ ▾ · 回复 · 分享 ›

---

**Pkme0119** · 3个月前

极好地文章

0 ^ ▾ · 回复 · 分享 ›

---

**Ryderkun** · 4个月前

i like it !

0 ^ ▾ · 回复 · 分享 ›

---

**Devafree** · 4个月前

good, favorite

0 ^ | v · 回复 · 分享 ›

jl2005 · 4个月前

信息量很大，需要慢慢消化。

0 ^ | v · 回复 · 分享 ›

Shuaiyang Yu · 4个月前

好文章

0 ^ | v · 回复 · 分享 ›

泉鹏 郭 · 4个月前

Thank you, I like github! I'll try it...;-)

0 ^ | v · 回复 · 分享 ›



robertsms · 4个月前

好文章，内容十分丰富，非常感谢！

0 ^ | v · 回复 · 分享 ›

kuke · 4个月前

zan

0 ^ | v · 回复 · 分享 ›

ADoyle · 5个月前

阅读过很多技术博客，但像这般写作质量如此高的真是很难得，钦佩博主的细心。

0 ^ | v · 回复 · 分享 ›

tracysu · 5个月前

ThumbUp

0 ^ | v · 回复 · 分享 ›

ringtail · 5个月前

0 ^ | v · 回复 · 分享

0 ^ | v · 回复 · 分享

0 ^ | v · 回复 · 分享

0 ^ | v · 回复 · 分享 ·

0 ^ | v · 回复 · 分享 ›



 通过邮件订阅