

git 本地仓库

从入门到工作：入门阶段

版权声明

本内容版权属杭州饥人谷教育科技有限公司（简称饥人谷）所有。

任何媒体、网站或个人未经本网协议授权不得转载、链接、转贴，或以其他方式复制、发布和发表。

已获得饥人谷授权的媒体、网站或个人在使用时须注明「资料来源：饥人谷」。

对于违反者，饥人谷将依法追究其责任。

联系方式

如果你想要购买本课程

请微信联系 [xiedaimala02](#) 或 [xiedaimala03](#)

如果你发现有人盗用本课程

请微信联系 [xiedaimala02](#) 或 [xiedaimala03](#)

你一定听过 GitHub

全球最大同性交友平台



Search GitHub

月.独.交.俊, 真人】约||会.

Explore

Help



Alex Gaynor

想跟他约的人

Washington D.C.
alex.gaynor@gmail.com
http://alexgaynor.net
Joined on Feb 25, 2008

1.8k Followers 154 Starred 41 Following

Organizations



Contributions

Repositories

Public activity

个人房间

他人房间

Popular repositories

what-happens-when	7,517 ★
An attempt to answer the age old interview q...	
django-taggit	1,213 ★
Simple tagging for django	
django-filter	1,060 ★
A generic system for filtering Django QueryS...	
django-ajax-validation	244 ★
A reusable application to preform ajax validat...	
django-templatetag-sugar	207 ★
This project exists to make defining template t...	

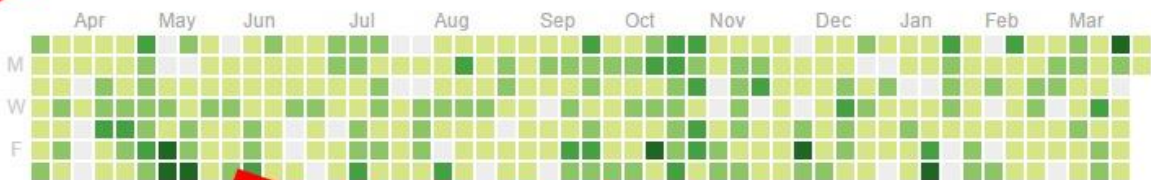
好评数

Repositories contributed to

pyca/cryptography	622 ★
cryptography is a package designed to expos...	
WhiteHouse/https	26 ★
The proposed HTTPS-Only Standard for fede...	
gowdprince/patch	74 ★
Homebrew/homebrew	22,038 ★
The missing package manager for OS X.	
PyCon/pycon	27 ★
Website for PyCon	

他想约的人

Public contributions



约会活动日历

Summary of Pull Requests, issues opened, and commits. Learn more

Less More

Contributions in the last year
2,481 total
Mar 23, 2014 – Mar 23, 2015

Longest streak
51 days
September 14 – November 3

Current streak
6 days
March 18 – March 23

git 就是开启 GitHub 的钥匙

想一起来同性交友吗？先学会 git 吧！

所有软件都是要先配置的

记不记得我说过这句话

git 的 6 行配置

请粘贴到命令行里运行

```
git config --global user.name 你的英文名  
git config --global user.email 你的邮箱  
git config --global push.default simple  
git config --global core.quotepath false  
git config --global core.editor "code --wait"  
git config --global core.autocrlf input
```

注意：上面的英文名和邮箱跟 GitHub 没有关系。
可以跟 GitHub 的用户名和邮箱保持一致，
也可以不一致。我的是一致的。

注意：你需要保证 code 是可以直接在命令行执行的。
如果不能执行，你需要安装 VSCode 并配置 PATH。
我的 PATH 里添加的内容是

C:\Users\Fang\AppData\Local\Programs\Microsoft VS Code\bin

运行完这六行
你才能开始用git

git 就是一个命令而已

Bash 命令行里有很多命令

git 解决了什么问题

一个只有程序员会遇到的问题：版本控制

考虑场景1

老板：给我做一个页面

程序员：花了一天，做出来了，请过目

老板：不够醒目，再改改

程序员：好嘞

程序员：又花了一天，请过目

老板：还是之前的好看，改回去

程序员：好的（MMP，不早说，我现在已经没法撤销了！）

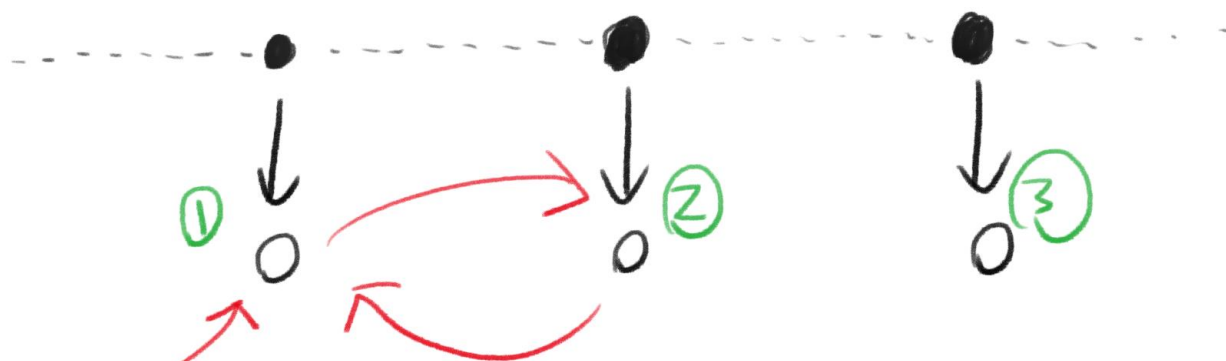
每次给老板看之前复制一下 不就好了

这是一个有效的土办法，很多公司就是这么做的

你的代码

复制的

给老板看



git 可以让你的代码有版本

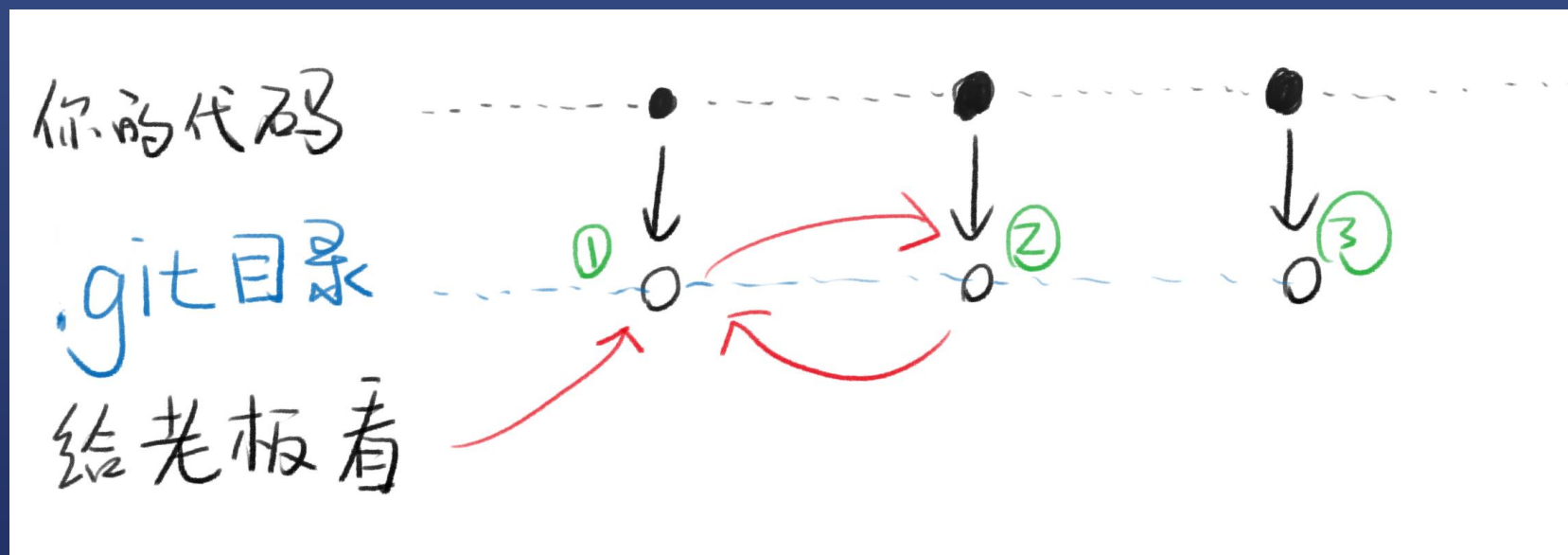
你可以随时回退到某个版本

当然 git 还有其他更强大的功能

git init

初始化

git init 会创建 .git 目录



git add 路径

选择哪些变动是需要提交的

路径可以是绝对路径

相对路径

. 和 *

.gitignore

描述哪些变动是不需要提交的

常见的有 node_modules

.DS_Store

.idea

.vscode

git commit -m 字符串

提交，并说明提交理由

字符串里如果有空格，就要用引号包起来

git commit -v

我个人更喜欢 --verbose 选项

因为它能帮我回顾我刚刚改了什么东西
而且会迫使我把提交理由写得更详细一些

我建议新人学习我这个习惯

已经创建了一个拷贝

可以通过 `git log` 查看

多提交几次

`git add .`

`git commit -v`

每次重复这两个操作即可

不要少写 .

不要少写空格

准备好御剑飞行了吗？



```
git reset --hard XXXXXX
```

XXXXXX 是提交号的前 6 位

请一定要确保你已经把所有代码 commit 了
因为这个操作会使没有 commit 过的变动消失

飞完了， 我现在想回去走访么办

git log 只会显示之前的提交

使用 git reflog 可以查看所有提交

然后用 git reset --hard XXXXXX 回去

总结

- git 可以实现多版本切换
 - ✓ git add 选择要提交的内容
 - ✓ .gitignore 文件描述不提交的内容
 - ✓ git commit -v 用来提交
 - ✓ git log 用来查看历史
 - ✓ git reset --hard XXXXXX 用来御剑飞行切换版本
 - ✓ git reflog 用来查看所有历史
- 少年，你已经学会穿越了
 - ✓ 就差一个女主角女朋友了

考虑场景2

老板：给我做一个页面

程序员：花了一天，做出来了，请过目

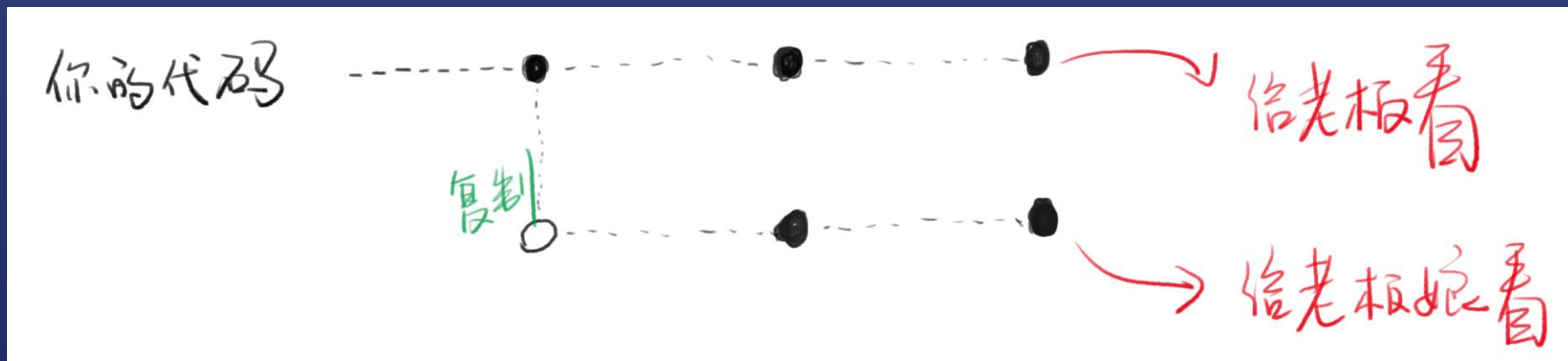
老板：不够醒目，再改改

老板娘：我觉得挺好，要是能做五彩斑斓的黑就好

程序员：好嘞，我做两份你们对比下（MMP，你们俩就不能统一意见吗！）

如何同时两条线开发？

土办法是复制一份，变成两个目录



git branch x

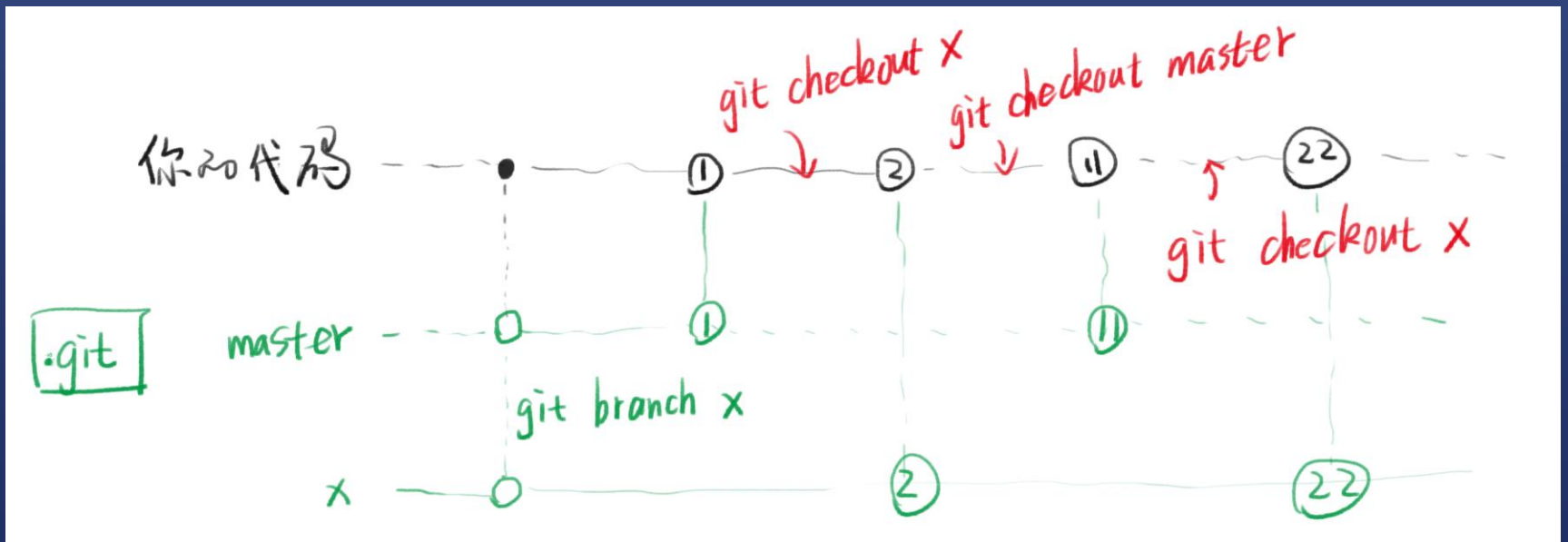
可以创造平行时间线 x

术语叫做「分支」

git branch x

git checkout x

两个时间线交替进行



总结

- git branch

- ✓ 基于当前 commit 创建一个新的时间线（分支）
- ✓ 我在哪个分支提交，代码就出现再哪个分支

- git checkout

- ✓ 用于切换另一个分支
- ✓ 当前目录有未提交的代码，只要跟另一个分支不冲突，就不需要理会
- ✓ 如果冲突了呢？可以使用通灵术 git stash，也可以合并冲突，后面遇到再细说

考虑场景3

老板：昨天夜里我和老板娘达成一致了，两个版本都要，我的放在在上面，老板娘在下面

程序员：好的，我合并一下

git merge

将另一个分支合并到当前分支

有可能有冲突
也可能没冲突

你要学会解决冲突

解决冲突的办法

• 发现冲突

- ✓ 你在合并分支的时候，会得到 conflict 提示
- ✓ 使用 `git status -sb` 查看哪个/哪些文件冲突了

• 解决冲突

- ✓ 依次打开每个文件
- ✓ 搜索 `====` 四个等于号
- ✓ 在上下两个部分中选择要保留的代码
- ✓ 可以只选上面，也可以只选下面，甚至可以都选
- ✓ 删除不用的代码，删除 `==== >>> <<<<` 这些标记
- ✓ `git add` 对应文件
- ✓ 再次 `git status -sb`，解决下一个文件的冲突
- ✓ 直到没有冲突，运行 `git commit`（注意不需要选项）

总结

- 分支可以合并

- ✓ 进入要保留的分支
- ✓ 运行 `git merge x`
- ✓ 合并完后删除无用的分支 `git branch -d x`

- 合并时冲突怎么办

- ✓ 解决冲突即可
- ✓ 然后 `git commit`, 使用默认的 `commit message`

大总结

- 我们学到的命令

- ✓ git config
- ✓ git add 路径
- ✓ git status -sb
- ✓ git commit -v
- ✓ git branch x
- ✓ git checkout x
- ✓ git merge
- ✓ git commit
- ✓ git branch -d x
- ✓ git log
- ✓ git reflog
- ✓ git reset --hard XXXXXX

大总结（续）

- .git 目录就是本地仓库

- ✓ 它不会重复复制相同的文件（优化）
- ✓ 它可以支持多个分支

- 一些细节

- ✓ git add 处理的是文件变化，而不是文件，比如你删除一个文件后，依然要用 git add 来添加到待提交区
- ✓ 大部分时候，你只需要知道 `git add .` 和 `git commit -v` 即可，其他命令很少用到
- ✓ 学会命令行操作，你就不再需要用 GUI 操作了
- ✓ 你工作之后，你的同事可能只会 GUI 操作，不要鄙视他们，他们没有时间重新打基础

关于 git 本地仓库就讲到这里

接下来看看 git 远程仓库