

【js 工具】如何在Github Pages搭建自己写的页面？ - 李佳怡 - 博客园

最近发现 github 改版了，已没有像原来的 Launch automatic page generator 这样的按钮等，所以我对我的文章也进行了修正，对于新版来说，步骤更加简单了。欢迎享用。

—————华丽丽的分割线，以下是在原版的基础上的修正版—————

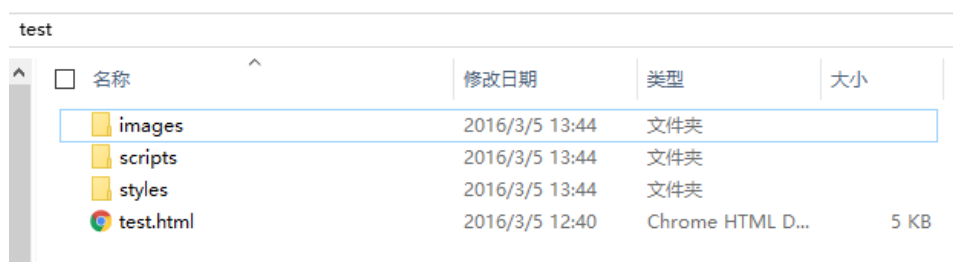
学了前端小半年，如今写了个自己的网页想要去应聘，却发现部署很麻烦，部署到阿里云之类，买域名啊啥的还要收费，说贵也不贵，但我就是傲娇~

google一下了解到Github有一个Github pages的功能可以搭建博客或者托管网页，而且免费耶，搜了下教程，猛地一看感觉步骤也不是很麻烦，所以就用这个了！

教程一大堆，却没有几个能看懂的，问题一：90%的都在讲解如何搭建博客，和我想要将自己的网页部署到上面还是有点区别的。问题二：所有的教程都用到了Git，而我只知道Git是一个开源的分布式版本控制系统。完全不知道命令行是什么鬼，只能照猫画虎的小白来说，那些教程只能帮我到桥头，但想要成功过河，还需要深夜里的一包特浓咖啡。

开始教程之前的准备工作：

1、需要你自己写的网页文件。



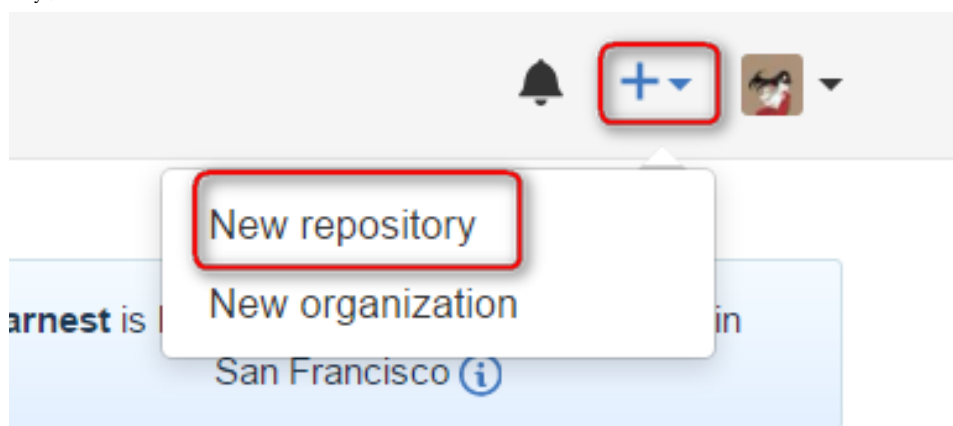
名称	修改日期	类型	大小
images	2016/3/5 13:44	文件夹	
scripts	2016/3/5 13:44	文件夹	
styles	2016/3/5 13:44	文件夹	
test.html	2016/3/5 12:40	Chrome HTML D...	5 KB

2、注册Github。

3、下载安装git。下载地址<https://git-scm.com/downloads>

教程开始：（以下出现的test指你的网页名或者你想起的一切名字）

步骤一：登录到Github上，新建一个repo，命名为test，勾选 initialize this repository with a README，点击 create repository。



Owner: JiayiLi

Repository name: test

Great repository names are short and memorable. Need inspiration? How about **curly-fiesta**.

Description (optional)

☒ Public
Anyone can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

☒ Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None

Add a license: None

Create repository

步骤二：打开settings，有一个Github Pages 的设置，点击 source 中的本来的 None ，使其变成 master 分支，也就是作为部署github pages 的分支，然后点击 save。

JiayiLi / test

Code Issues Pull requests Wiki Pulse Graphs Settings

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Source

GitHub Pages is currently disabled. Select a source below to enable GitHub Pages for this repository. [Learn more.](#)

master branch Save

Select source

- ☒ master branch
Use the master branch for GitHub Pages.
- ☐ master branch /docs folder
Use only the /docs folder for GitHub Pages.
- ☐ None
Disable GitHub Pages.

第二步

第一步

步骤三：页面刷新之后，再看 github pages 设置框处，多了一行网址，就是你的 github pages 的网址了。

GitHub Pages

✓ Your site is published at <https://jiayili.github.io/test-/>

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Source

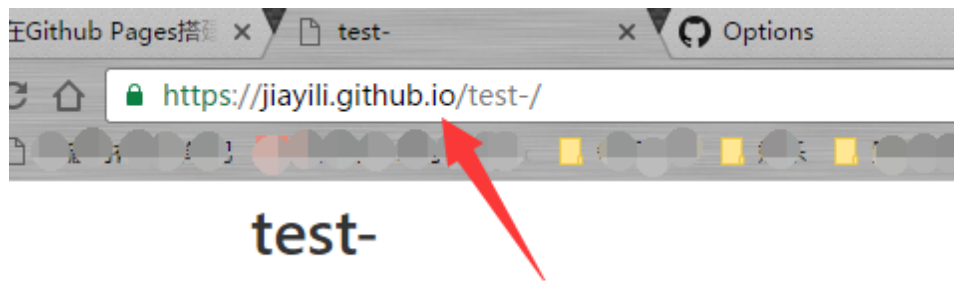
Your GitHub Pages site is currently being built from the master branch. [Learn more.](#)

master branch Save

Theme chooser

Select a theme to build your site with a Jekyll theme. [Learn more.](#)

点击这个链接

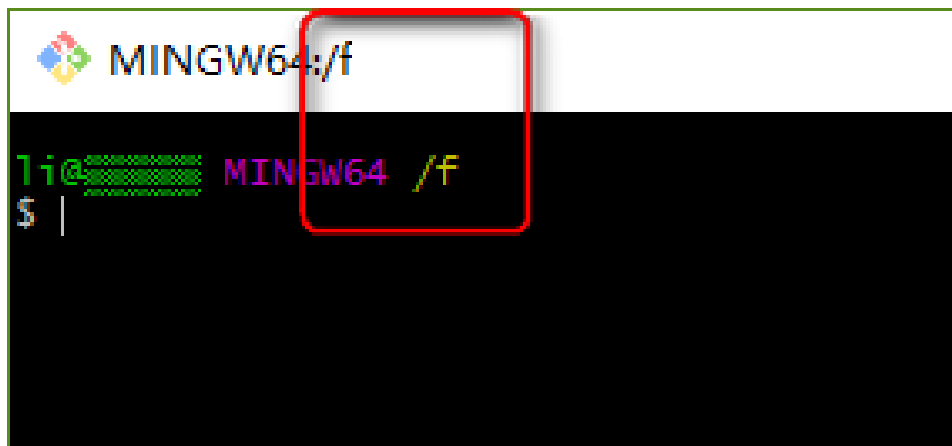


哇塞，一个 test。

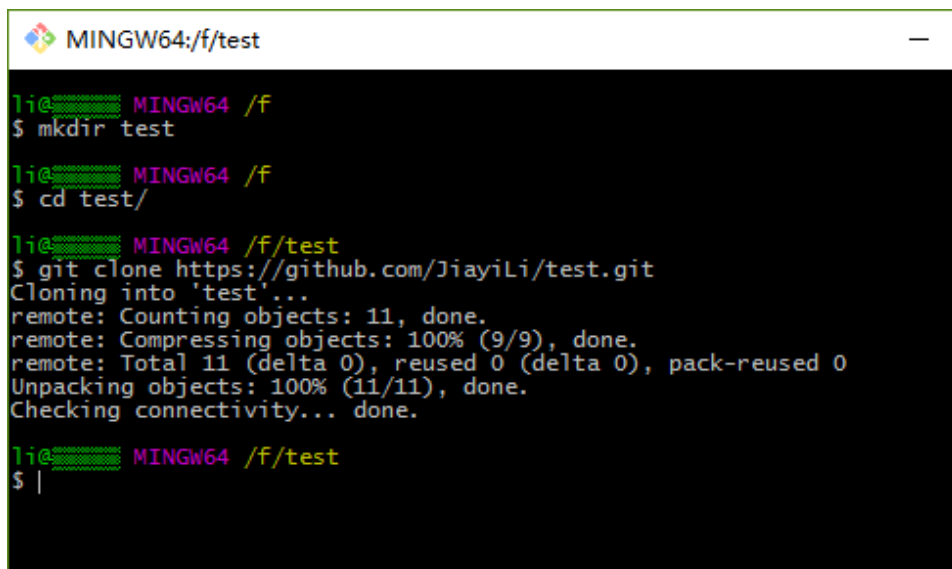
好了，别激动了，真正让一个小白你发疯的阶段开始了，有了自己的网址，但页面显示的是什么鬼？

至此以上，github上要处理的操作告一段落，该上Git了！

步骤四：打开此电脑，选择一个盘，比如 f 盘，右键空白处点击 git bash here。



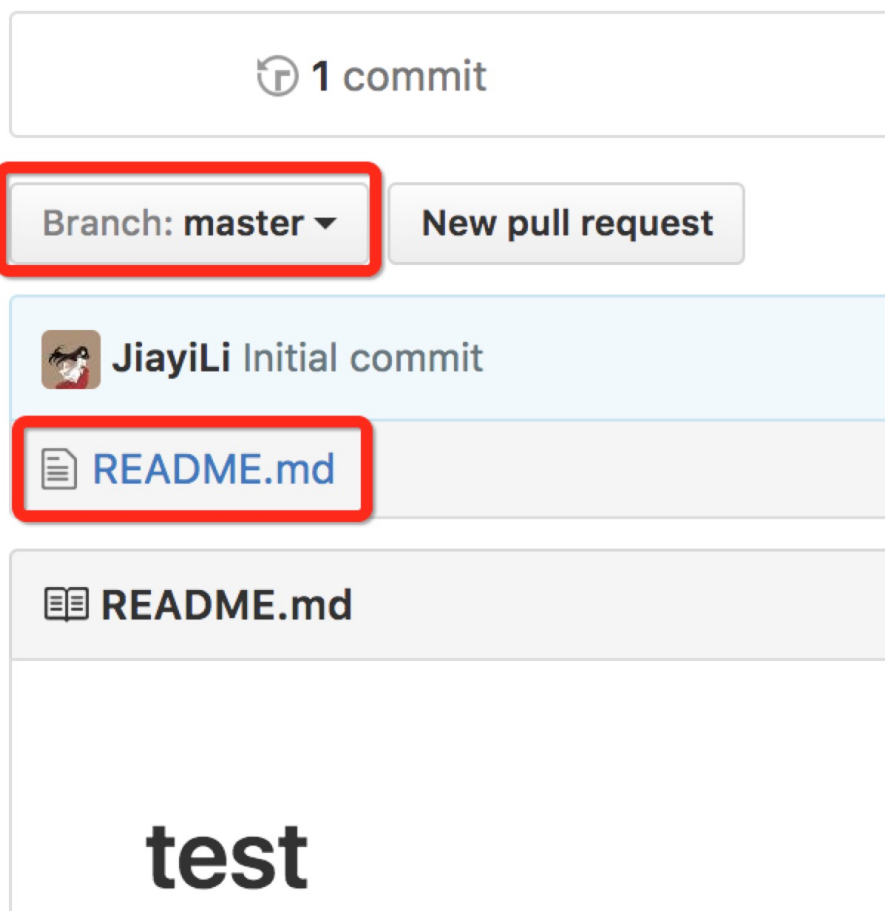
步骤五：输入如下命令，用来在 f 盘创建 test 文件夹放你的github上的test repository，克隆test repository到 test 文件夹中。



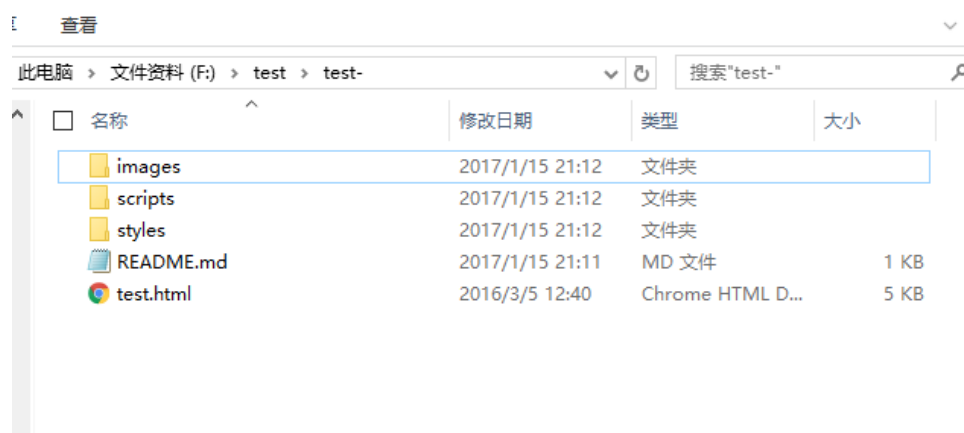
这个时候你的 f 盘，就会多一个 test 文件夹，打开它，



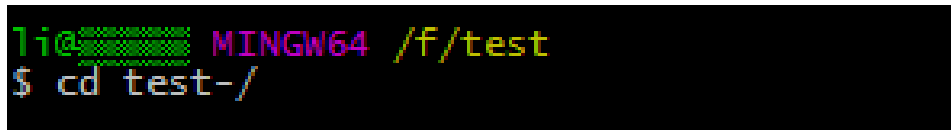
会看到一个 README.md 的文件，这个文件是从哪来的呢？追溯到github上，你会发现 README 文件是来自 master 分支。



步骤六： 将自己的网页文件复制粘贴至 f 盘的 test 文件中



步骤八： 执行如下命令



```

li@MINGW64 /f/test/test- (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    images/
    scripts/
    styles/
    test.html

nothing added to commit but untracked files present (use "git add" to track)
li@MINGW64 /f/test/test- (master)
$ git add .
warning: LF will be replaced by CRLF in scripts/jquery.js.
The file will have its original line endings in your working directory.
li@MINGW64 /f/test/test- (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   images/1.png
    new file:   images/2.png
    new file:   images/3.png
    new file:   images/4.png
    new file:   images/about.jpg
    new file:   images/bg.jpg
    new file:   images/close.png
    new file:   images/edu.jpg
    new file:   images/envelope.png
    new file:   images/favicon.jpg
    new file:   images/firstPage.jpg
    new file:   images/loading.gif
    new file:   images/mobile.png
    new file:   images/nav.png
    new file:   images/prize.jpg
    new file:   images/rel.jpg
    new file:   images/search.png
    new file:   images/skills.jpg
    new file:   images/three_stripes_horiz_533px_1160221_easyicon.net.png
    new file:   scripts/jquery.js
    new file:   scripts/loading.js
    new file:   scripts/resumeaboutme.js
    new file:   scripts/topbarnav.js
    new file:   styles/resume.css
    new file:   test.html
li@MINGW64 /f/test/test- (master)

```

解释一下上面的命令：首先输入 `git status` 列出当前目录所有还没有被git管理的文件和被git管理且被修改但还未提交(`git commit`)的文件，也就是所有改动文件，红色字体标出。

然后输入 `git add .`（有个点）表示添加当前目录下的所有文件和子目录，

然后 再输入一次 `git status` 如果看见文件都变绿了，那么就代表 它们已经准备好了被提交（`git commit`）。

步骤九：输入如下命令，将你的文件上传至远程 `master` 分支

```

li@MINGW64 /f/test/test- (master)
$ git commit -m "modify"
[master 6b18411] modify
warning: LF will be replaced by CRLF in scripts/jquery.js.
The file will have its original line endings in your working directory
25 files changed, 11809 insertions(+)
create mode 100644 images/1.png
create mode 100644 images/2.png
create mode 100644 images/3.png
create mode 100644 images/4.png
create mode 100644 images/about.jpg
create mode 100644 images/bg.jpg
create mode 100644 images/close.png
create mode 100644 images/edu.jpg
create mode 100644 images/envelope.png
create mode 100644 images/favicon.jpg
create mode 100644 images/firstPage.jpg
create mode 100644 images/loading.gif
create mode 100644 images/mobile.png
create mode 100644 images/nav.png
create mode 100644 images/prize.jpg
create mode 100644 images/rel.jpg
create mode 100644 images/search.png
create mode 100644 images/skills.jpg
create mode 100644 images/three_stripes_horiz_533px_1160221_easy
create mode 100644 scripts/jquery.js
create mode 100644 scripts/loading.js
create mode 100644 scripts/resumeaboutme.js
create mode 100644 scripts/topbarnav.js
create mode 100644 styles/resume.css
create mode 100644 test.html

li@MINGW64 /f/test/test- (master)
$ git pull
Already up-to-date.

li@MINGW64 /f/test/test- (master)
$ git push
warning: push.default is unset; its implicit value has changed in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the traditional behavior, use:

    git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

    git config --global push.default simple

When push.default is set to 'matching', git will push local branches
to the remote branches that already exist with the same name.

Since Git 2.0, Git defaults to the more conservative 'simple'
behavior, which only pushes the current branch to the corresponding
remote branch that 'git pull' uses to update the current branch.

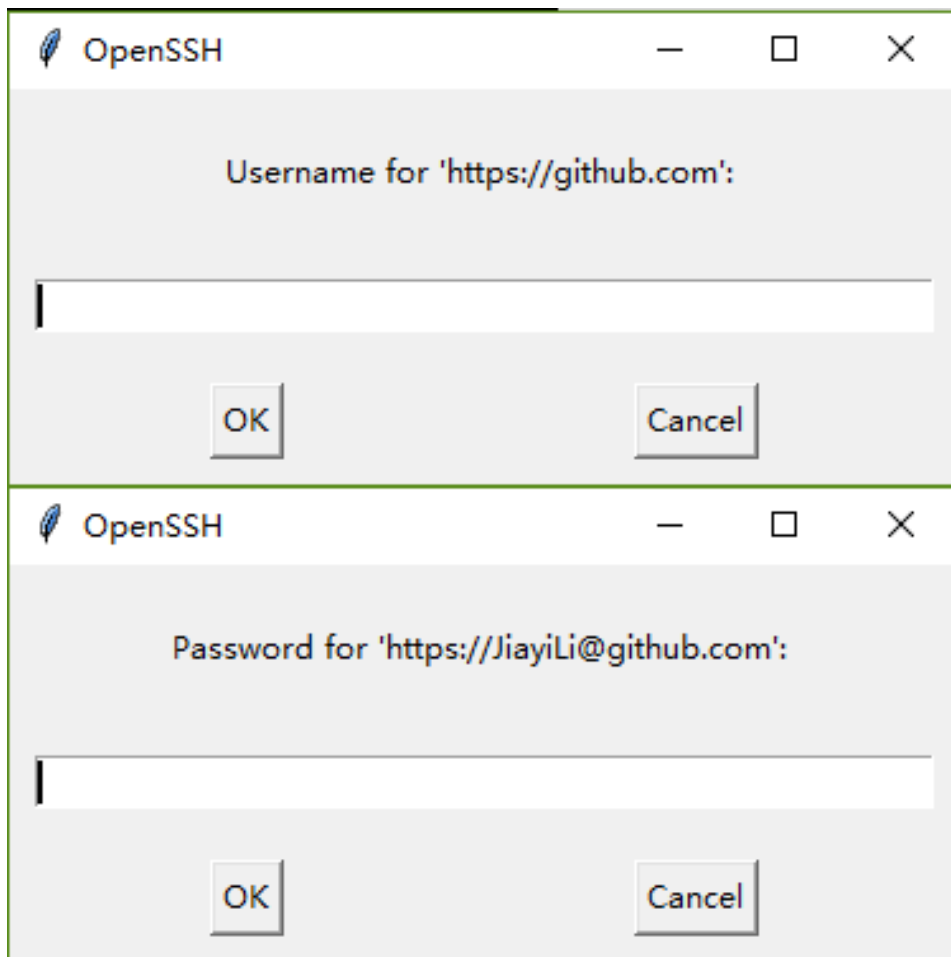
See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Counting objects: 30, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (29/29), done.
Writing objects: 100% (30/30), 2.05 MiB | 232.00 KiB/s, done.
Total 30 (delta 0), reused 0 (delta 0)
To https://github.com/JiayiLi/test-.git
c658c62..6b18411 master -> master

li@MINGW64 /f/test/test- (master)
$ git status

```

输入最后一行 git push, 按回车, 等一会, 会有弹出框让你输入你的 github 账号和密码。

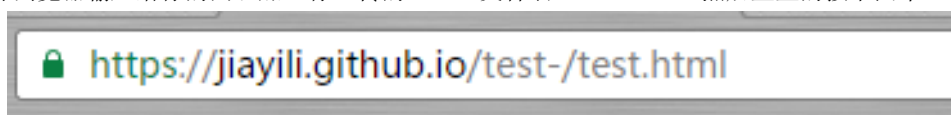


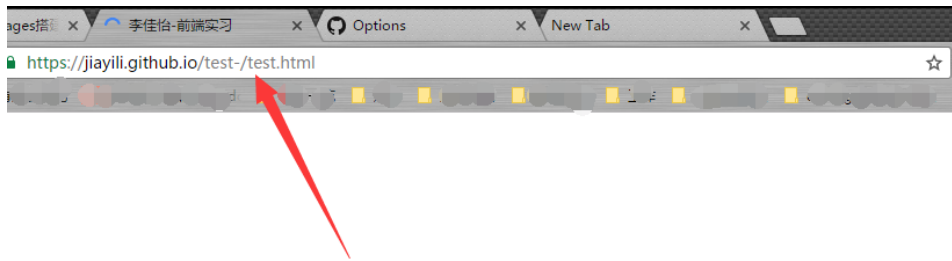
ok之后耐心等待。

当出现像下图中的语句之后，你已经完成了99%。

```
Counting objects: 30, done.  
Delta compression using up to 4 threads.  
Compressing objects: 100% (29/29), done.  
Writing objects: 100% (30/30), 2.05 MiB | 232.00 KiB/s, done.  
Total 30 (delta 0), reused 0 (delta 0)  
To https://github.com/Jiayili/test-.git  
c658c62..6b18411 master -> master
```

最后一步：打开浏览器输入给你的网址加上你上传的 html 文件名 test.html，然后重重的按下回车。





恭喜你，成功了！！！哦不，还差一步，不关注一下我？不推荐一下文章？

附录一：常用git命令：

```
$ git clone //本地如果无远程代码，先做这步，不然就忽略
$ cd //定位到你blog的目录下
$ git status //查看本地自己修改了多少文件
$ git add . //添加远程不存在的git文件
$ git commit -m "what I want told to someone" //提交修改
$ git push //更新到远程服务器上
$ git rm //移除文件
```

附录二：如何修改你的网页？

方法一：在github上的master分支中直接进行修改。（如果不添加新的文件推荐此法）

方法二：在github客户端进行修改在同步。（如果添加了新的文集推荐此法）

推荐阅读：

- 1、【js 算法类】这么全的数组去重，你怕不怕？ <http://www.cnblogs.com/lijiayi/p/shuzuquchong.html>
- 2、如何使用 Git 上传本地项目到 github ? (mac 版) <http://www.cnblogs.com/lijiayi/p/pushtogithub.html>
- 3、【js 性能优化】underscore throttle 与 debounce 节流 <http://www.cnblogs.com/lijiayi/p/jsoptimisel.html>
- 4、【js 基础】深浅拷贝 <http://www.cnblogs.com/lijiayi/p/jsdeeepcopy.html>

最后的最后 推荐关注我的前端知乎专栏 TristaLee的纸糊 <https://zhuanlan.zhihu.com/front-end-web>