



搭建一个自己的 BLOG



A circular inset image showing a cityscape at sunset or sunrise. The sun is low on the horizon, casting a golden glow over the buildings. A prominent building in the center has 'Allianz' written on its top. The foreground shows a street with cars and lower-rise buildings.

为什么要写博客?

-
1. 记录学习过程。
 2. 减少二次学习带来的麻烦。
 3. 分享到公共平台的博客有时也可以给别人带来帮助

关于GitHub

1.使用前了解Github

GitHub是一个共享虚拟主机服务，用于存放使用Git版本控制的软件代码和内容项目(引自维基百科)

2.为什么选择Github?

github有一个很有爱的项目，叫做github pages，这个项目是给开发者建立一个私人页面，上面用来分享新颖的想法和自己写的代码。

开始搭建BLOG

主要介绍

1、搭建博客的大致流程。2、过程中使用到的命令。3、换电脑后如何转移博客



建立博客的基本流程

一、环境准备

- 1.安装node.js
- 2.安装git
- 3.注册 Github账号

二、配置

- 1.安装Hexo
- 3.部署到Github (ssh key)
- 4.主题

三、写作

- 1.Markdown语法
- 2.html语法



视作到移居某地

一、环境准备

1.了解当地人的语言

3.找到住所

二、配置

1.学习当地语言、向该地提交自己的信息

3.配置住所的钥匙

4.装饰房子，适应环境。

三、后续的学习生活



环境准备

配置Node.js环境

(如果是window10系统可以装64-bit)

- [Windows Installer 32-bit](#)
- [Windows Installer 64-bit](#)

保持默认设置即可，一路Next

打开命令行界面 (win+R—输入cmd—回车) 输入命令。

使用到的命令

- **node -v**
- **npm -v**

可以看到所下载的版本 (版本可能不同)

[教程参考](#)

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.18362.418]
(c) 2019 Microsoft Corporation。保留所有权利。

C:\Users\13204>node -v
v10.16.3

C:\Users\13204>npm -v
6.9.0

C:\Users\13204>
```

环境准备

配置Git环境

(如果是window10系统可以装64-bit)

Git官网下载地址:

保持默认设置即可，一路Next

打开命令行界面输入命令。

使用到的命令

- **git --version**

可以看到所下载的版本（版本可能不同）

[教程参考](#)

```
MINGW64:/c/Users/13204

13204@DESKTOP-AF0UJID MINGW64 ~
$ git --version
git version 2.20.1.windows.1

13204@DESKTOP-AF0UJID MINGW64 ~
$
```


环境准备

注册 Github 账号

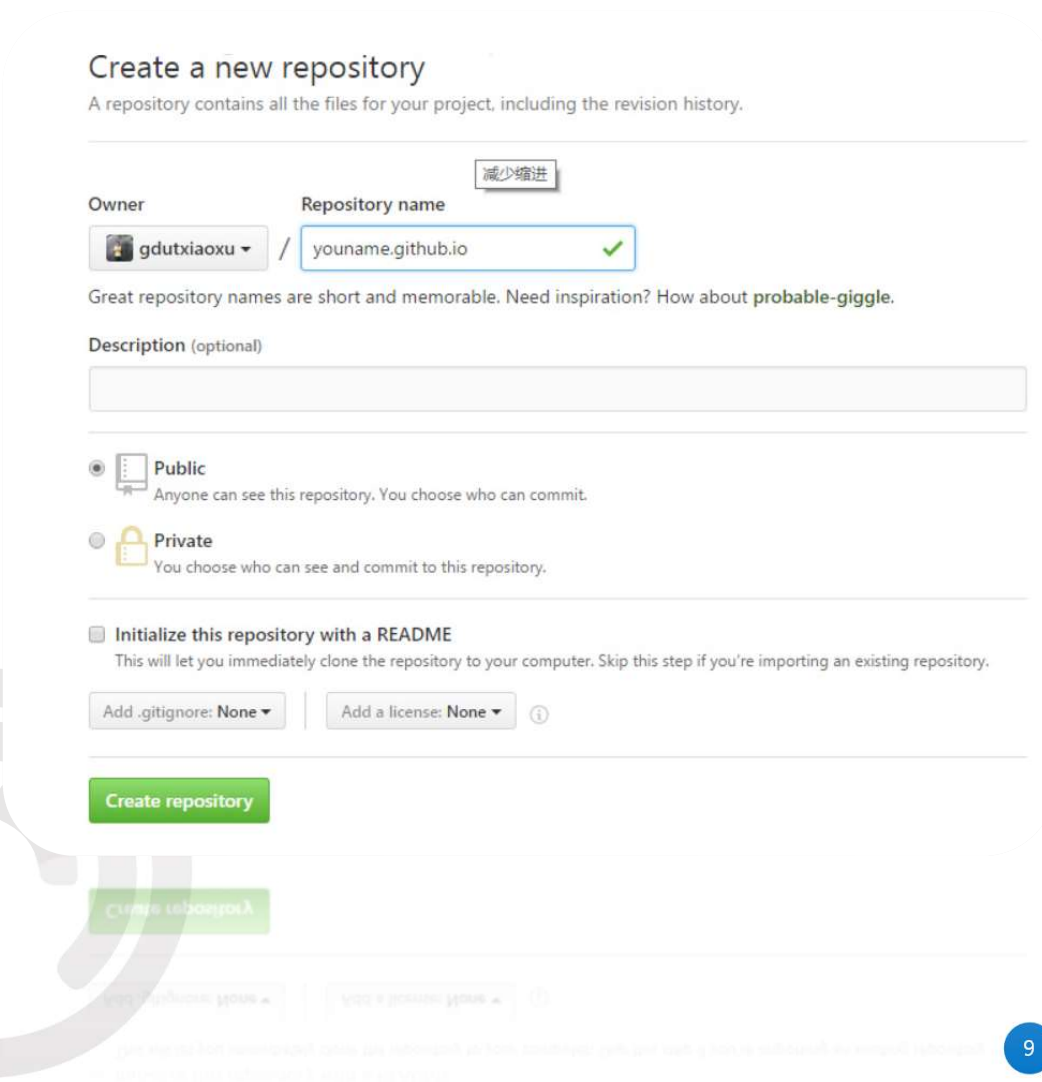
- 在github网址 上注册账号。
- 注册后需要在填写的邮箱进行验证。
- 已有账号可以略过此步。

点击页面右上角的加号，选择New repository
如图填写信息

易错点：

“yourname”指的是注册时写的用户名

[教程参考](#)



Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: / Repository name:

Great repository names are short and memorable. Need inspiration? How about **probable-giggle**.

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: | Add a license:

环境准备

开启gh-pages功能

1. 点击界面右侧的Settings从而打开这个库的setting页面
2. 向下拖动，直到看见GitHub Pages
3. 点击Automatic page generator

完成后Github将会自动替你创建一个gh-pages的页面。

如果你的配置没有问题，那么大约15分钟之后yourname.github.io这个网址就可以正常访问了~

[教程参考](#)

Overwrite site

Replace your existing site by using our automatic page generator. Author you select a theme, then publish.

Launch automatic page generator



安装HEXO

1. 安装



1. 在认为合适的地方创建一个文件夹
2. 打开git命令窗口 (右键—Git Bash Here)
3. 输入
 - `npm install hexo-cli -g` (载入hexo command)
 - `npm install hexo -save`
 - `hexo -v` (检查版本, 看看是否安装好)

```
E:\hexo>hexo -v
hexo: 3.2.2
hexo-cli: 1.0.2
os: Windows_NT 6.3.9600 w
http_parser: 2.5.0
node: 4.2.3
v8: 4.5.103.35
uv: 1.7.5
zlib: 1.2.8
ares: 1.10.1-DEV
icu: 56.1
modules: 46
openssl: 1.0.2e
```

接着上面的操作
输入

- `hexo g`
- `hexo s`

出现提示

INFO Hexo is running at <http://0.0.0.0:4000/>. Press Ctrl+C to stop.

在浏览器中打开<http://localhost:4000/>
就可以看到自己的blog页面了



2. HEXO的相关配置

部署到GITHUB

配置GIT个人信息 (SSH)



1、生成密钥

- `ssh-keygen -t rsa -C gdutxiaoxu@163.com`

2、找到系统当前用户目录下(开启查看隐藏文件) `C:\Users\用户名\.ssh id_rsa.pub`文件以文本方式打开。打开之后全部复制到key中。

3、测试一下是否成功了:

输入

`ssh -T git@github.com`

#之后会要你输入yes/no,输入yes就好了。



The screenshot illustrates the steps to configure an SSH key on GitHub. It is divided into three numbered sections:

- 1. Profile Menu:** Shows the GitHub profile dropdown menu with the 'Settings' option highlighted by a red arrow.
- 2. SSH Keys List:** Shows the 'SSH keys' page in the GitHub settings. A red arrow points to the 'New SSH key' button in the top right corner.
- 3. Add SSH Key Form:** Shows the 'Add SSH key' form. A red arrow points to the 'Key' text area, which contains the instruction: 'id_rsa.pub 文件下的复制到这' (Copy from the id_rsa.pub file to here). Another red arrow points to the 'Add SSH key' button at the bottom of the form.

1

2

3

12

设置你的账号信息

输入:

- `git config --global user.name "你的名字"`
- `git config --global user.email "邮箱@邮箱.com"`

#user为你的用户名

部署到GITHUB

输入:

- `hexo d`

这时再刷新 `username.github.io` 就可以看到你的博客了。



4.设计页面

1、下载主题



- 1、网页搜索“hexo主题”，选择喜欢的主题下载到theme文件夹下
- 2、配置文件 config.xml中把theme换成下载的主题文件夹的名字，
- 3、而后进入下载主题的文件夹，可以看到里面也有一个配置文件 _config.xml，貌似它默认是 _config.xml.example，把它复制一份，重命名为 config.xml就可以了。这个配置文件是修改你整个主题的配置文件的



Fan

A dark theme

#中文 #dark #Pug #Stylus



```
git clone https://github.com/fan-lv/Fan.git the
```

修改位于博客根目录下的 _config.yml 内的 theme 选

更新

```
cd themes/Fan  
git pull
```

需要的依赖包

- hexo-renderer-jade
- hexo-renderer-stylus

问题

- 如果报错 appid undefined,需要把 themes->_co

说明

- 支持文章置顶功能，在需要置顶的文章加入 top:

License

MIT

在下载的主题那里仔细看作者一起发布的一些内容，那里会有一些关于使用主题的贴士，例如需要安装（git install...）一些函数。



SO ON

[更多参考](#)

写作——建议学习顺序

- Markdown语法（快速上手）
- HTML css建议看书
- JS语法 建议看书

学习

换电脑后转移文件

设置你的账号信息

输入:

- `git config --global user.name "你的名字"`
- `git config --global user.email "邮箱@邮箱.com"`

#user为你的用户名

如果发现错误找不到原因可以删除文件重来一遍，文件管理和扫描一样是一个好习惯。

步骤



补充步骤

skycrown
Android开发

174 人赞同了该回答

不知道题主是不是换了新电脑，需要在新电脑上进行部署，如果是，可以参考下面的方法：
1. 从官网Git下载git，在新电脑上安装，因为https速度慢，而且每次都要输入口令，常用的是使用ssh，使用下面方法创建：
(1) 打开git bash，在用户主目录下运行：`ssh-keygen -t rsa -C "youremail@example.com"`
把其中的邮件地址换成自己的邮件地址，然后一路回车
(2) 最后完成后，会在用户主目录下生成ssh目录，里面有id_rsa和id_rsa.pub两个文件，这两个就是SSH key密钥对，id_rsa是私钥，千万不能泄露出去，id_rsa.pub是公钥，可以放心地告诉任何人。
(3) 登陆GitHub，打开「Settings」->「SSH and GPG keys」，然后点击「new SSH key」，填上任意title，在Key文本框里粘贴公钥id_rsa.pub文件的内容（千万不要粘贴成私钥！），最后点击「Add SSH Key」，你就应该看到已经添加的Key。
注意：不要在git版本库中运行ssh，然后又将它提交，这样就把密码泄露出去了。
2. 下载Node.js，并安装
3. 打开git bash客户端，输入 `npm install hexo-cli -g`，开始安装hexo
4. 下面就将原来的文件拷贝到新电脑中，但是要注意哪些文件是必须的，哪些文件是可以删除的。
(1) 讨论下哪些文件是必须拷贝的：首先是之前自己修改的文件，像站点配置_config.yml，theme文件夹里面的主题，以及source里面自己写的博客文件，这些肯定是要拷贝的。除此之外，还有三个文件需要，就是scaffolds文件夹（文章的模板）、package.json（说明使用哪些包）和gitignore（限定在提交的时候哪些文件可以忽略）。其实，这三个文件不是我们修改的，所以即使丢失了，也没有关系，我们可以建立一个新的文件来，然后在里面执行hexo init，就会生成这三个文件，我们只需要将它们拷贝过来使用即可。**总结：**_config.yml, theme/, source/, scaffolds/, package.json, .gitignore, 是需要拷贝的。
(2) 再讨论下哪些文件是不必拷贝的，或者说可以删除的：首先是.git文件，无论是在站点根目录下，还是主题目录下的.git文件，都可以删掉，然后是文件夹node_modules（在使用npm install会重新生成），public（这个在用hexo g时会重新生成），.deploy_git文件夹（在使用hexo d时也会重新生成），db.json文件。其实上面这些文件也就是.gitignore文件里面记载的可以忽略的内容。**总结：**.git/, node_modules/, public/, .deploy_git/, db.json文件需要删除。
5. 在git bash中切换目录到新拷贝的文件夹里，使用 `npm install` 命令，进行模块安装。很明显我们这里没用hexo init初始化，因为有的文件我们已经拷贝生成过来了，所以不必用hexo init去整体初始化，如果不慎在此时用了hexo init，则站点的配置文件_config.yml里面内容会被清空使用默认值，所以这一步一定要慎重，不要用hexo init。
6. 安装其他的一些必要组件，如果在node_modules里面有的，就不要重复安装了：
(1) 为了使用hexo d来部署到git上，需要安装
`npm install hexo-deployer-git --save`
(2) 为了建立RSS订阅，需要安装
`npm install hexo-generator-feed --save`
(3) 为了建立站点地图，需要安装
`npm install hexo-generator-sitemap --save`
插件安装后，有的需要对配置文件_config.yml进行配置，具体怎么配置，可以参考上面插件在github主页上的具体说明
7. 使用hexo g，然后使用hexo d进行部署，如果都没有出错，就转移成功了！

「广告时间」：如果觉得有用，不妨点个「赞」，让更多人看到。如果想转载，请注明出处。
^_^!
编辑于 2018-07-07



谢谢观看
