

typora使用



目录

typora使用

目录

1、typora添加到鼠标右键选项

2、添加typora主题

3、关联git与github（关联后可将文件传输到GitHub仓库中）

github部分

公钥添加

创建一个专门存图片和文档的库

git部分

下载

git通用的工作流程

设置

创建并查看公钥

通过git控制github的库

4、typora图传github

设置PicGo-Core

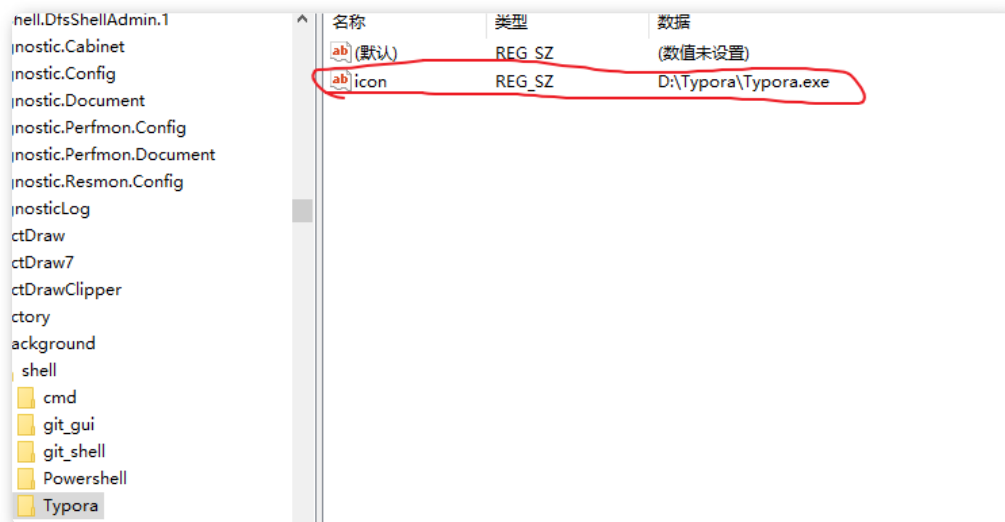
将PicGo与github账户的仓库相关联

1、typora添加到鼠标右键选项

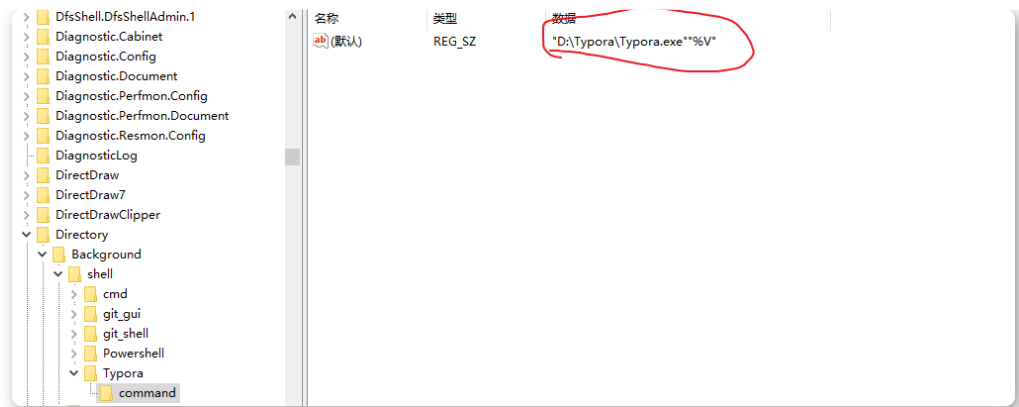
本文链接：

/ [2、添加typora主题](#)/ / [3、关联git与github](#)/ / [4、typora图传github](#)/

1. 打开注册表：win+r,输入regedit
2. 定位到 HKEY_CLASSES_ROOT\Directory\Background\shell
3. 在刚才定位的位置下新建项Typora，在项中再新建一个字符串值，命名为icon,将其数据改为typora的安装路径






4. 右键Typora新建项command，填写默认值的数据路径："D:\TyporaTypora.exe""%V"



2、添加typora主题



本文链接：

/  1、typora添加到鼠标右键选项 //  3、关联git与github/ /  4、typora图传github/




"文件"-->"偏好设置"-->"外观"-->"获取主题"


将主题的.css文件放在typora的主题文件夹中，重启typora，便可在"主题"选项中选择添加的主题。

可以在主题作者的指导下，更改主题中的颜色和字体。

3、关联git与github（关联后可将文件传输到GitHub仓库中）

本文链接：


/  1、typora添加到鼠标右键选项 //  2、添加typora主题//  4、typora图传github/

参考：  Typora+Github | 神仙笔记软件 | 版本管理 | Markdown哔哩哔哩bilibili

 git基本操作，一篇文章就够了！ - 掘金 (juejin.cn)

github部分

公钥添加

如果在电脑中第一次使用git一定要记得在本地  创建公钥，并添加到github中，保证本地和远端的github是可以连接的。以下为具体步骤：

GitHub官网-->"settings"-->"SSH and GPG keys"-->"new SSH Key"

填入title后，将在Git Bash中获得的公钥复制进入key中，公钥就添加完成了。

■ 创建一个专门存图片和文档的库



■ git部分

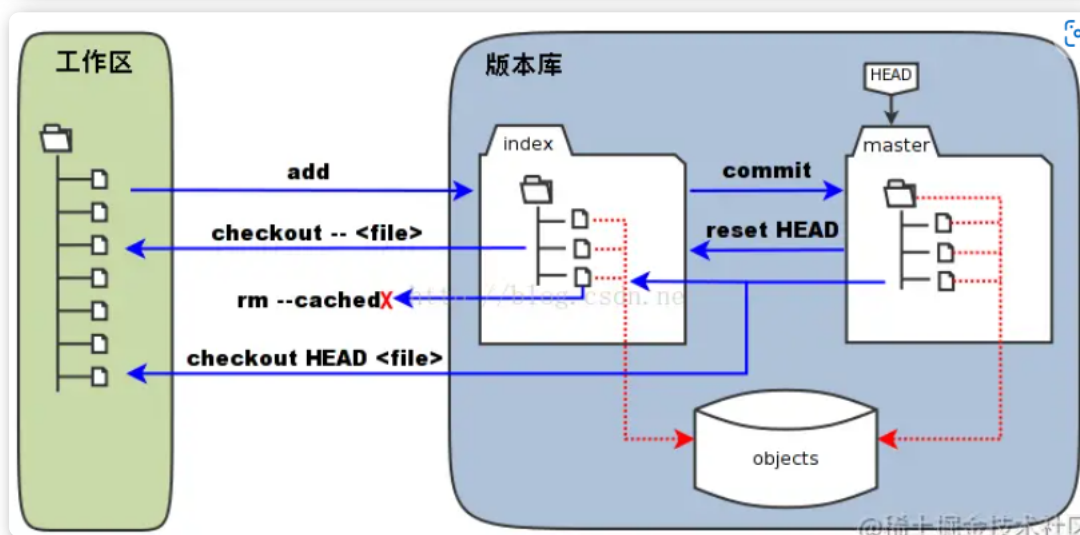
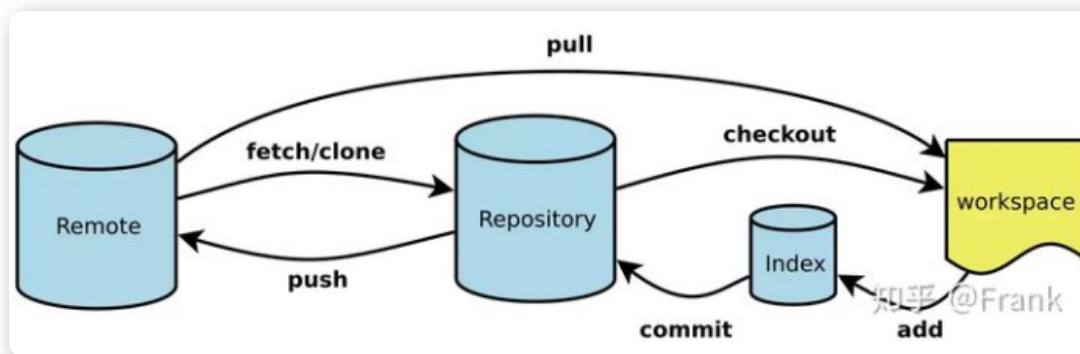


■ 下载



在 Git (git-scm.com)中下载git

■ git通用的工作流程



名称	作用
工作区 (workspace)	本地电脑存放项目文件的地方，比如learnGitProject文件夹
暂存区 (Index/Stage)	在使用git管理项目文件的时候，其本地的项目文件会多出一个.git的文件夹，将这个.git文件夹称之为版本库。其中.git文件夹中包含了两个部分，一个是暂存区（Index或者Stage），顾名思义就是暂时存放文件的地方，通常使用add命令将工作区的文件添加到暂存区里
本地仓库 (Repository)	.git文件夹里还包括git自动创建的master分支，并且将HEAD指针指向master分支。使用commit命令可以将暂存区中的文件添加到本地仓库中
远程仓库 (Remote)	不是在本地仓库中，项目代码在远程git服务器上，比如项目放在github上，就是一个远程仓库，通常使用clone命令将远程仓库拷贝到本地仓库中，开发后推送到远程仓库中即可

设置



● 创建并查看公钥



- 1、在Git Bash输入以下命令创建密钥（中间提示要设置的密码时可以不设置，直接enter）：`ssh-keygen -o`

```

34265@LAPTOP-313BMKQI MINGW64 /repository/first-repository
$ ssh-keygen -o
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/34265/.ssh/id_rsa):
Created directory '/c/Users/34265/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/34265/.ssh/id_rsa
Your public key has been saved in /c/Users/34265/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:ZiT/LpHtgCURTwE3BB55fDTqtjug/v9CE4VNvzIaMOE 34265@LAPTOP-313BMKQI
The key's randomart image is:
+---[RSA 3072]-----+
|
|  +=. =+
| 000*.+0
| E+0+. .
|   ++. .
|   +BSo.
|  o.**++.
| . *o. +.
| . +....
| .....0+...
|
+-----[SHA256]-----+

```

2、输入以下命令查看密钥（此处获得的就是创建的公钥了）：`cat`
`/c/Users/34265/.ssh/id_rsa.pub`

注意：cat指令后的地址为公钥文件保存的地址，具体获得的方法在上图用红线标记

```

34265@LAPTOP-313BMKQI MINGW64 /repository/first-repository
$ cat /c/Users/34265/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDsKvTiyVX5T5eok5gKKN42vshZFU0sVyvNtaItJakXawUoKQoQN0AdR33aciu8RRr
MFNkmt88/MR/ehe8C+yyrnvEu//lJdc21lNeQ/tjTNn5ASQdjI1u9W3Z56DTIM8Q2dJKsNsFhZ2gov8etw3UwLPPT9QAy10wBeAMEFG
d9DUBz5PNxno/DYrHi1GgH2BrWZQ9+13C0cCXIAJeDtq9ehN/F0oA6S9dVhqpEPuTAGmqf0rocmjIiSes1y8pqYP82bKHL8Lus2Zdz1
xazF9Z9VfdnEoPvxy5YkeBav+ObGHIS2X423ztT5g7wolk5ogrdfMS+1JiiuBh8IOFztga9DICqwk4QbS1C276BF8sJq4p4qmw2gZY8
OCnrvXqXiPTL6uWdQ2s9A/MF1abTqOXLYFH502ecIx1LNnZnWFTPJB5WaXJMdcaYgo10/Mi9RTi+MB4u7MTkC+5Abj0E5aXpw4tKjii
sCG07WN2CLYw1wJQKrrAYqwwvLXkMrVE20CU= 34265@LAPTOP-313BMKQI

```

● 通过git控制github的库



注意：必须在你决定好的本地库中打开Git Bash

```

1 //设置信息
2 git config --global user.name "Your Name" //Your
  Name:你的昵称
3 git config --global user.email "email@example.com"
  //email@example.com:你的邮箱
4
5 //将GitHub的库复制到本地库

```




```
6  git clone git@github.com:moutao-123/first-repository
   //将git clone指令后为库的连接
7
8  git pull origin main    //从远程的main分支获取后合并到你当前
   的工作区
9
10 //查看工作区相对于暂存区的差别
11 git status              //指令后再加上-s可以输出简短的结果
12
13 //更改库中文件的内容
14 cd picture             //移动到picture文件夹下
15 ls picture             //显示picture文件夹中的所有文件
16 mkdir A                //在工作区创建一个名为A的工作夹
17 touch A.txt            //如果目录下不存在A.txt,在工作区创建一个A.txt
   文件,若存在即为更改这个文件的时间戳
18 git add A.txt          //将A.txt从工作区添加到暂存区
19 git rm A.txt           //将A.txt从工作区和暂存区中删除
20
21 git commit -m '可以写入此次的改动' //将暂存区里的改动给提交到
   本地的版本库
22
23 git push <远程主机名> <本地分支名> <远程分支名> //将本地的
   分支推送到远程主机上的对应分支
24 /*
25  如果当前分支只有一个远程分支,那么主机名都可以省略,如 git
   push。可以使用git branch -r , 查看远程的分支名
26 */
```

通过以上命令可以简单的将本地的文件（包括typora建立的文件）传输到github中

4、typora图传github

本文链接：

/  1、typora添加到鼠标右键选项 //  2、添加typora主题//  3、关联git与github/

参考： Typora使用PicGo-Core（不用安装node.js）作为图片上传服务，并使用github作为图床哔哩哔哩bilibili


设置PicGo-Core

"打开typora" --> "文件" --> "偏好设置" --> "图像" --> "上传服务设定" 后面的选项卡中的 "PicGo-Core(command line)" --> "下载并更新"

将PicGo与github账户的仓库相关联

1. 在电脑文件中进入 "C:\Users\你电脑的用户名\AppData\Roaming\Typora\picgo\win64" 的路径
2. 以cmd命令打开当前文件夹
3. 在终端里输入 "picgo set uploader" 命令
4. 通过移动光标选择"github",然后回车
5. 在"repo:"后输入 "A"/B" (A: 自己的github的名字)(B : github中存图片的仓库的名字)
6. "branch:"后输入 "main"
7. "token:" (token的获取办法: github官网 --> "settings" --> "<>Developer setting" --> "personal access tokens" --> "Generate new token" --> 写上

Note的内容后，选中" Select scopes " 的 "repo" --> "Generate token" --> 复制绿色背景中的一串英文)

8. "customUrl:"后输入"  <https://cdn.jsdelivr.net/gh/你的github名字/你要存图片的仓库名>"
 9. 输入"picgo use uploader"命令
 10. 通过移动光标选择"github",然后回车
-