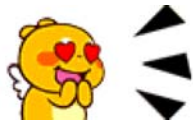


# Git的安装和使用（托管至GitHub的方法）

chestnut\_egg Python爱好者社区 2 days ago



点击蓝字 轻松关注



作者: chestnut\_egg Python爱好者社区专栏作者  
博客: <https://www.cnblogs.com/chestnut-egg>

## 一、下载Git

### 1. 下载

下载地址:

<https://git-scm.com/download/win>

根据你的操作系统选择32位或者64位



### Your download is starting...

You are downloading the latest (**2.16.2**) **64-bit** version of **Git for Windows**. This is the most recent **maintained build**. It was released **about 2 months ago**, on 2018-02-20.

If your download hasn't started, [click here to download manually](#).

### Other Git for Windows downloads

**Git for Windows Setup**

**32-bit Git for Windows Setup.**

**64-bit Git for Windows Setup.**

**Git for Windows Portable ("thumbdrive edition")**

**32-bit Git for Windows Portable.**

**64-bit Git for Windows Portable.**

The current source code release is version **2.17.0**. If you want the newer version, you can build it from [the source code](#).

## 2.安装过程一路next

## 3.检验安装是否成功

在桌面点击右键，如果出现这两个选项代表安装成功



## 二、Git初始化及使用

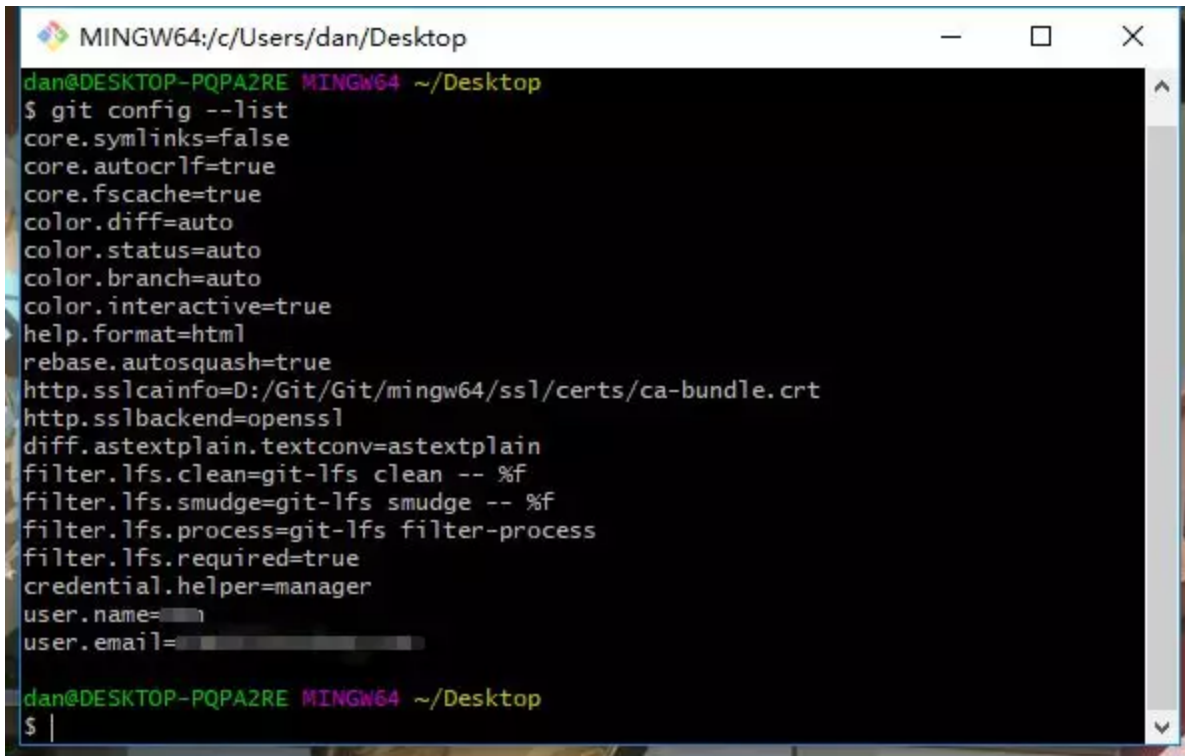
### 1.点击右键选择 Git Bash Here

输入以下命令：

```
git config -- global  user.name  '你的用户名';  
git config -- global  user.email  '你的邮箱';
```

### 2.配置后可以输入此条命令看是否设置成功

```
git config --list
```



```
MINGW64:/c/Users/dan/Desktop
dan@DESKTOP-PQPA2RE MINGW64 ~/Desktop
$ git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
rebase.autosquash=true
http.sslcainfo=D:/Git/Git/mingw64/ssl/certs/ca-bundle.crt
http.sslbackend=openssl
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
credential.helper=manager
user.name=[redacted]
user.email=[redacted]
dan@DESKTOP-PQPA2RE MINGW64 ~/Desktop
$ |
```

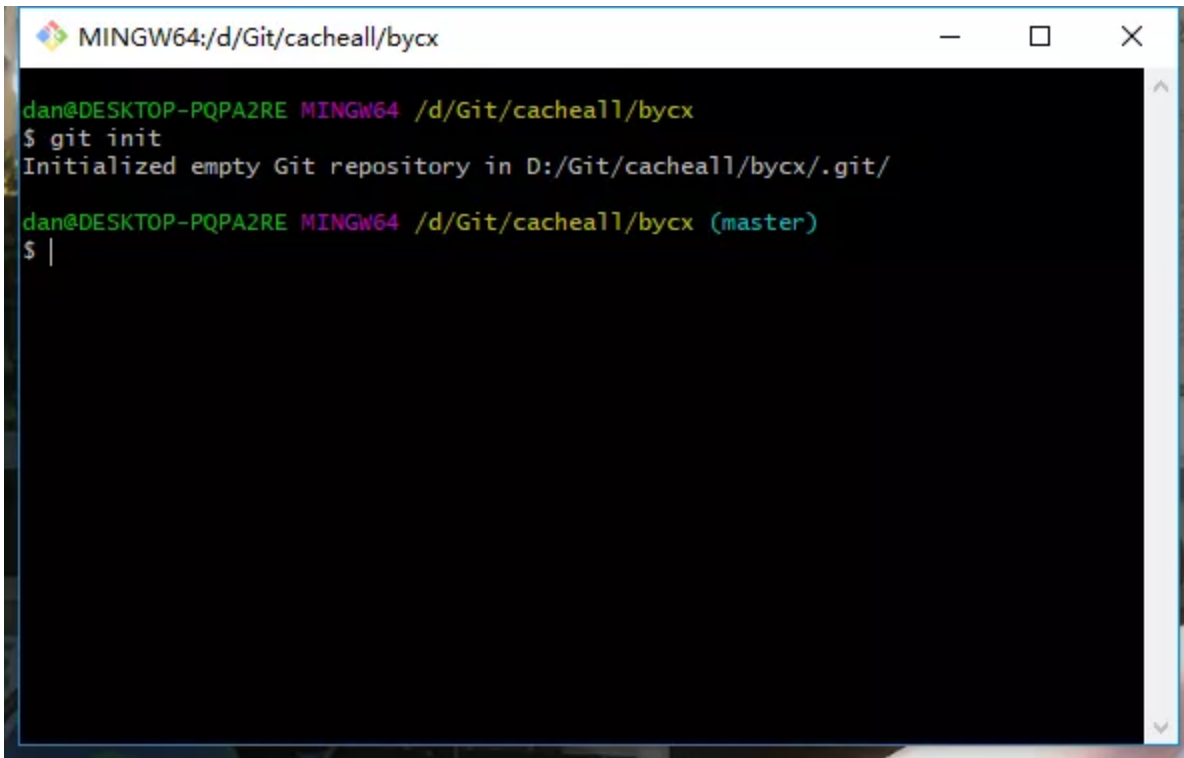
### 3. 创建本地仓库

创建一个文件夹，用作你的仓库

打开这个文件夹，在这个目录下右键，选择Git Bash Here

输入此条命令：

```
git init
```



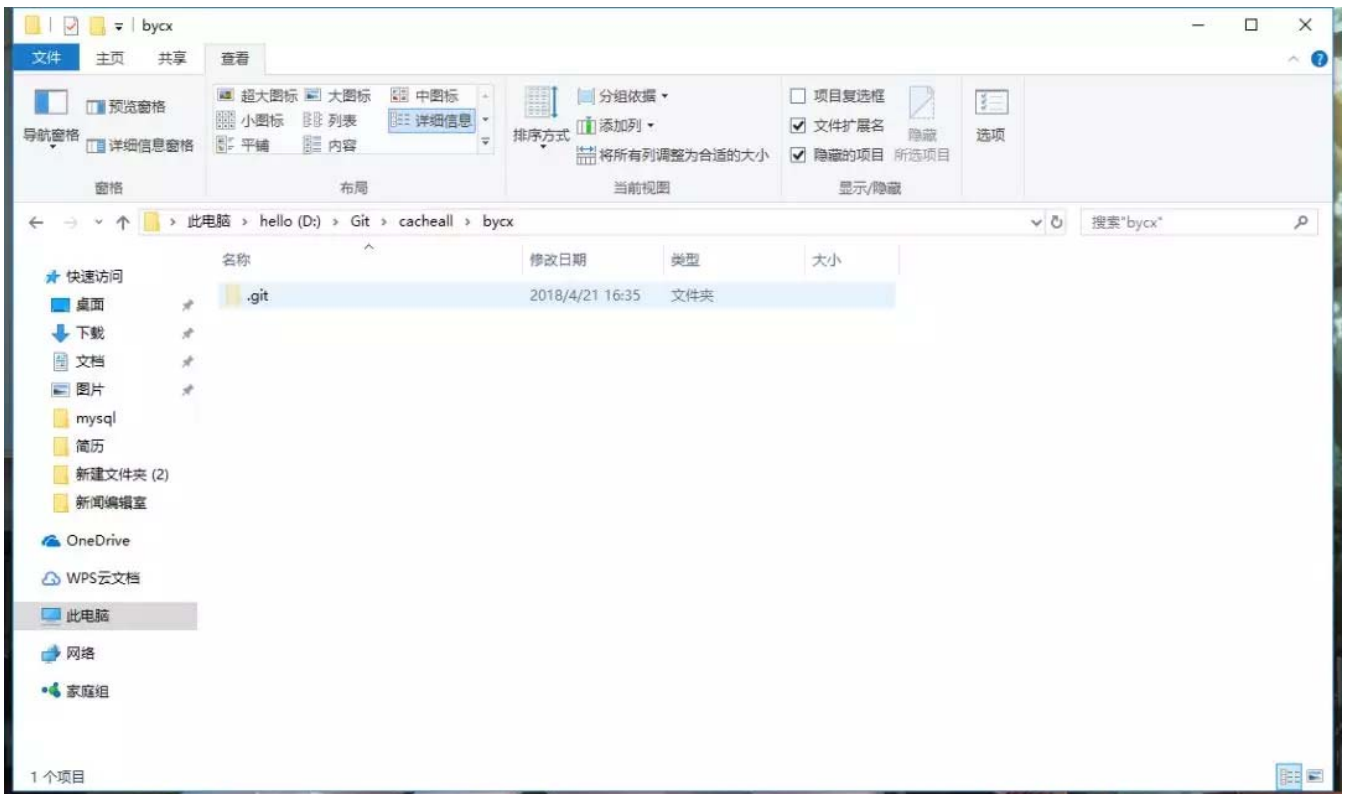
```
MINGW64:/d/Git/cacheall/bycx

dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx
$ git init
Initialized empty Git repository in D:/Git/cacheall/bycx/.git/

dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ |
```

4.然后文件夹中会出现一个.git的文件夹

该文件夹属性为隐藏

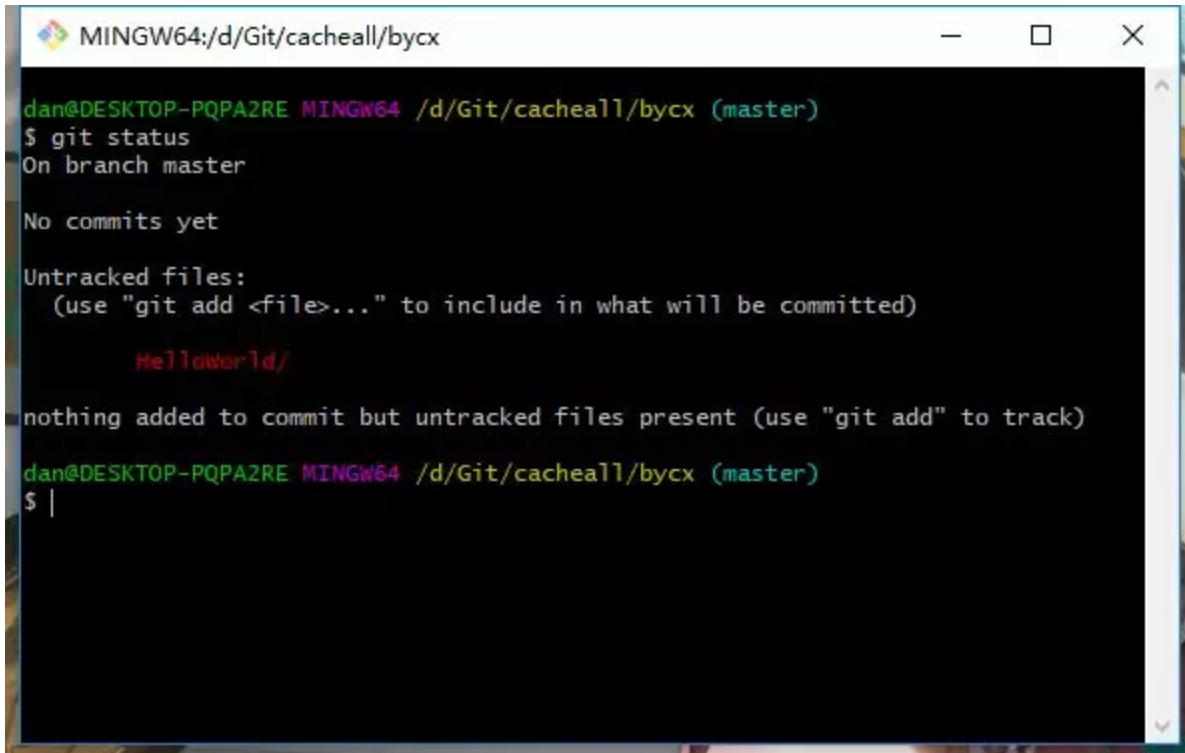


5.将你的项目复制到此文件夹中

## 6.将文件提交到缓冲区

执行以下命令

```
git status
```



```
MINGW64:/d/Git/cacheall/bycx
dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        HelloWorld/

nothing added to commit but untracked files present (use "git add" to track)
dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ |
```

发现飘红了，这是因为我们没有将文件提交到缓存区

使用这条命令可将所有文件提交到缓冲区

注意：add 与 . 之间有一个空格，代表目录下的所有文件，如果只提交一个文件，也可以输入这个文件的文件名

```
git add .
```

再次使用

```
git status
```

 MINGW64:/d/Git/cacheall/bycx

```
dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ git add .
warning: LF will be replaced by CRLF in HelloWorld/.idea/HelloWorld.iml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in HelloWorld/.idea/inspectionProfiles/profiles_settings.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in HelloWorld/.idea/misc.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in HelloWorld/.idea/modules.xml.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in HelloWorld/.idea/workspace.xml.
The file will have its original line endings in your working directory.

dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   HelloWorld/.idea/HelloWorld.iml
    new file:   HelloWorld/.idea/inspectionProfiles/profiles_settings.xml
    new file:   HelloWorld/.idea/misc.xml
    new file:   HelloWorld/.idea/modules.xml
    new file:   HelloWorld/.idea/workspace.xml
    new file:   HelloWorld/HelloWorld/__init__.py
    new file:   HelloWorld/HelloWorld/__pycache__/__init__.cpython-36.pyc
    new file:   HelloWorld/HelloWorld/__pycache__/settings.cpython-36.pyc
    new file:   HelloWorld/HelloWorld/__pycache__/urls.cpython-36.pyc
    new file:   HelloWorld/HelloWorld/__pycache__/view.cpython-36.pyc
    new file:   HelloWorld/HelloWorld/__pycache__/wsgi.cpython-36.pyc
    new file:   HelloWorld/HelloWorld/settings.py
    new file:   HelloWorld/HelloWorld/urls.py
    new file:   HelloWorld/HelloWorld/view.py
    new file:   HelloWorld/HelloWorld/wsgi.py
    new file:   HelloWorld/db.sqlite3
    new file:   HelloWorld/manage.py
    new file:   HelloWorld/templates/hello.html
    new file:   HelloWorld/templates/student_home.html

dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ |
```

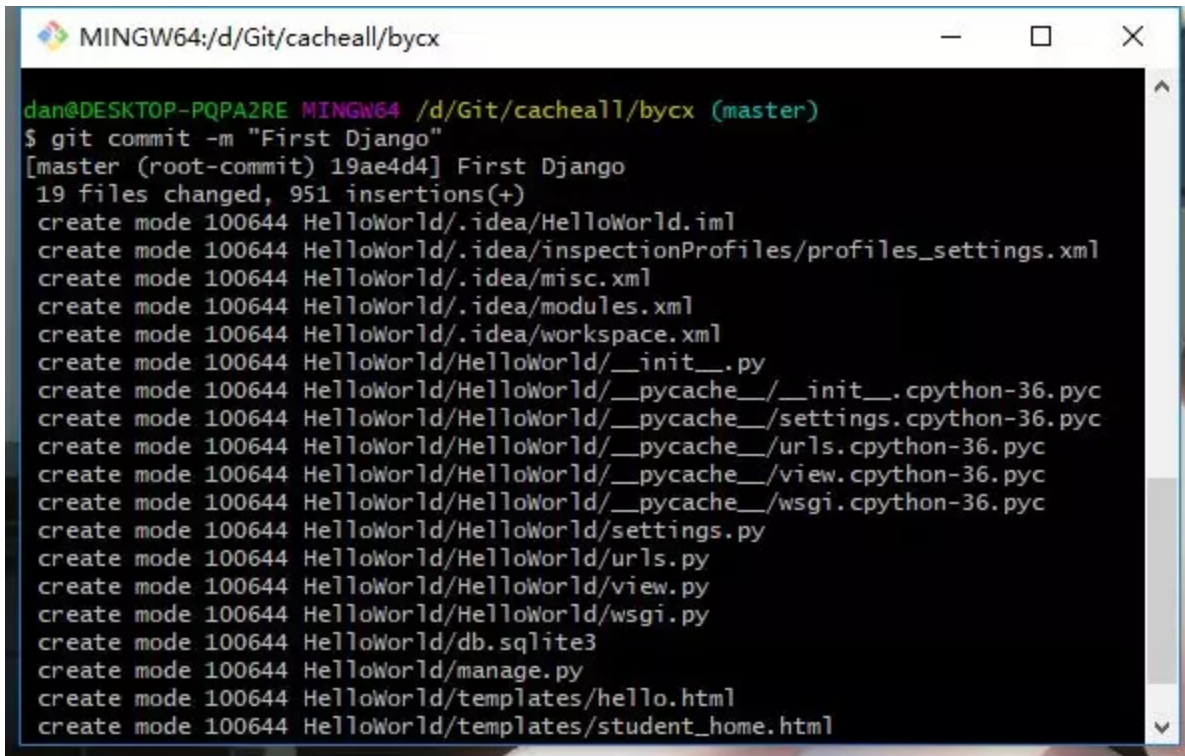
绿色就很舒服~

## 6.将缓冲区文件提交到本地仓库

输入命令:

```
git commit -m "你的备注"
```



A terminal window titled 'MINGW64:/d/Git/cacheall/bycx' with standard window controls. The prompt is 'dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)'. The command '\$ git commit -m "First Django"' has been executed. The output shows the commit hash '[master (root-commit) 19ae4d4] First Django', the number of files changed '19 files changed, 951 insertions(+)', and a list of 19 files created with mode '100644'. The files include IDE configuration files (.idea/HelloWorld.iml, .idea/inspectionProfiles/profiles\_settings.xml, .idea/misc.xml, .idea/modules.xml, .idea/workspace.xml), Python files (HelloWorld/\_\_init\_\_.py, \_\_pycache\_\_/\*.cpython-36.pyc, urls.cpython-36.pyc, view.cpython-36.pyc, wsgi.cpython-36.pyc, settings.py, urls.py, view.py, wsgi.py), and static files (db.sqlite3, manage.py, templates/hello.html, templates/student\_home.html).

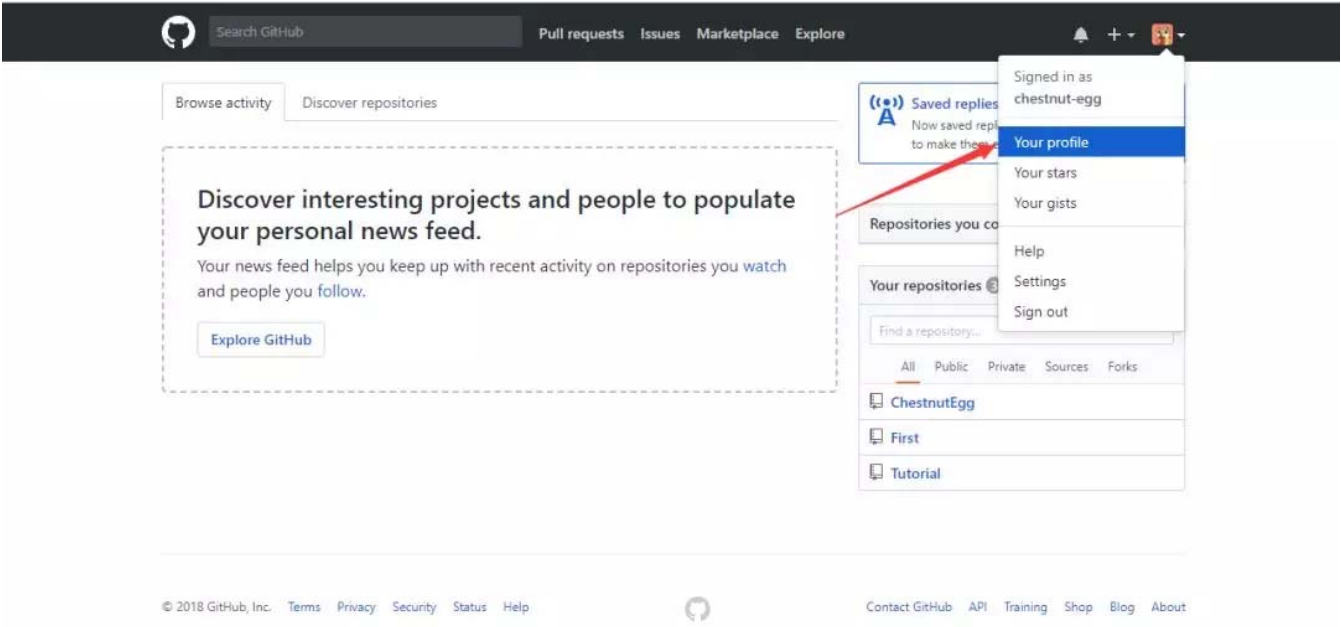
```
MINGW64:/d/Git/cacheall/bycx
dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ git commit -m "First Django"
[master (root-commit) 19ae4d4] First Django
19 files changed, 951 insertions(+)
create mode 100644 HelloWorld/.idea/HelloWorld.iml
create mode 100644 HelloWorld/.idea/inspectionProfiles/profiles_settings.xml
create mode 100644 HelloWorld/.idea/misc.xml
create mode 100644 HelloWorld/.idea/modules.xml
create mode 100644 HelloWorld/.idea/workspace.xml
create mode 100644 HelloWorld/HelloWorld/__init__.py
create mode 100644 HelloWorld/HelloWorld/__pycache__/__init__.cpython-36.pyc
create mode 100644 HelloWorld/HelloWorld/__pycache__/settings.cpython-36.pyc
create mode 100644 HelloWorld/HelloWorld/__pycache__/urls.cpython-36.pyc
create mode 100644 HelloWorld/HelloWorld/__pycache__/view.cpython-36.pyc
create mode 100644 HelloWorld/HelloWorld/__pycache__/wsgi.cpython-36.pyc
create mode 100644 HelloWorld/HelloWorld/settings.py
create mode 100644 HelloWorld/HelloWorld/urls.py
create mode 100644 HelloWorld/HelloWorld/view.py
create mode 100644 HelloWorld/HelloWorld/wsgi.py
create mode 100644 HelloWorld/db.sqlite3
create mode 100644 HelloWorld/manage.py
create mode 100644 HelloWorld/templates/hello.html
create mode 100644 HelloWorld/templates/student_home.html
```

## 二、将项目托管至GitHub

### 1.创建GitHub账号

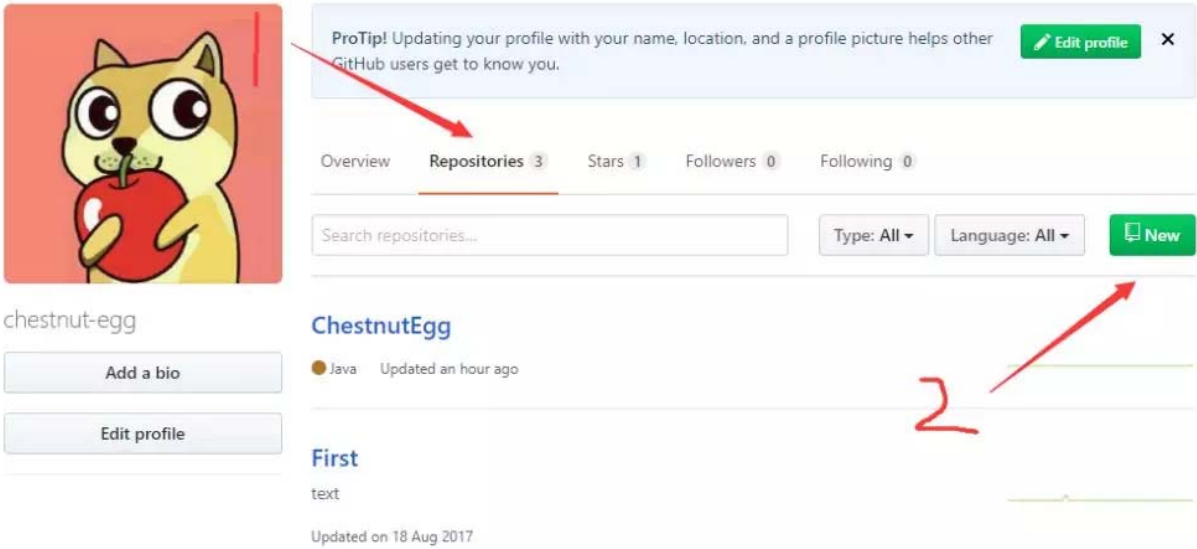
地址：<https://github.com/>

### 2.点击右上角头像，选择your profile

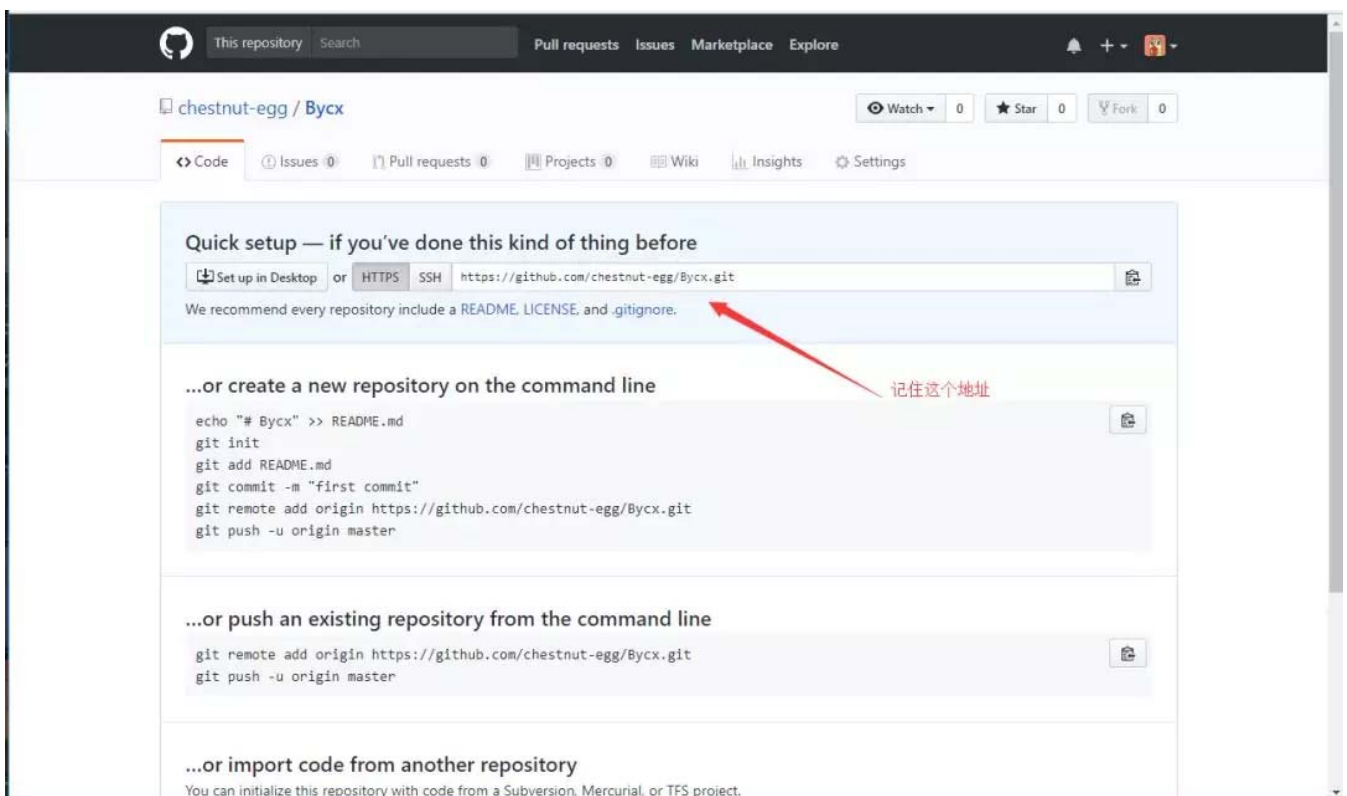
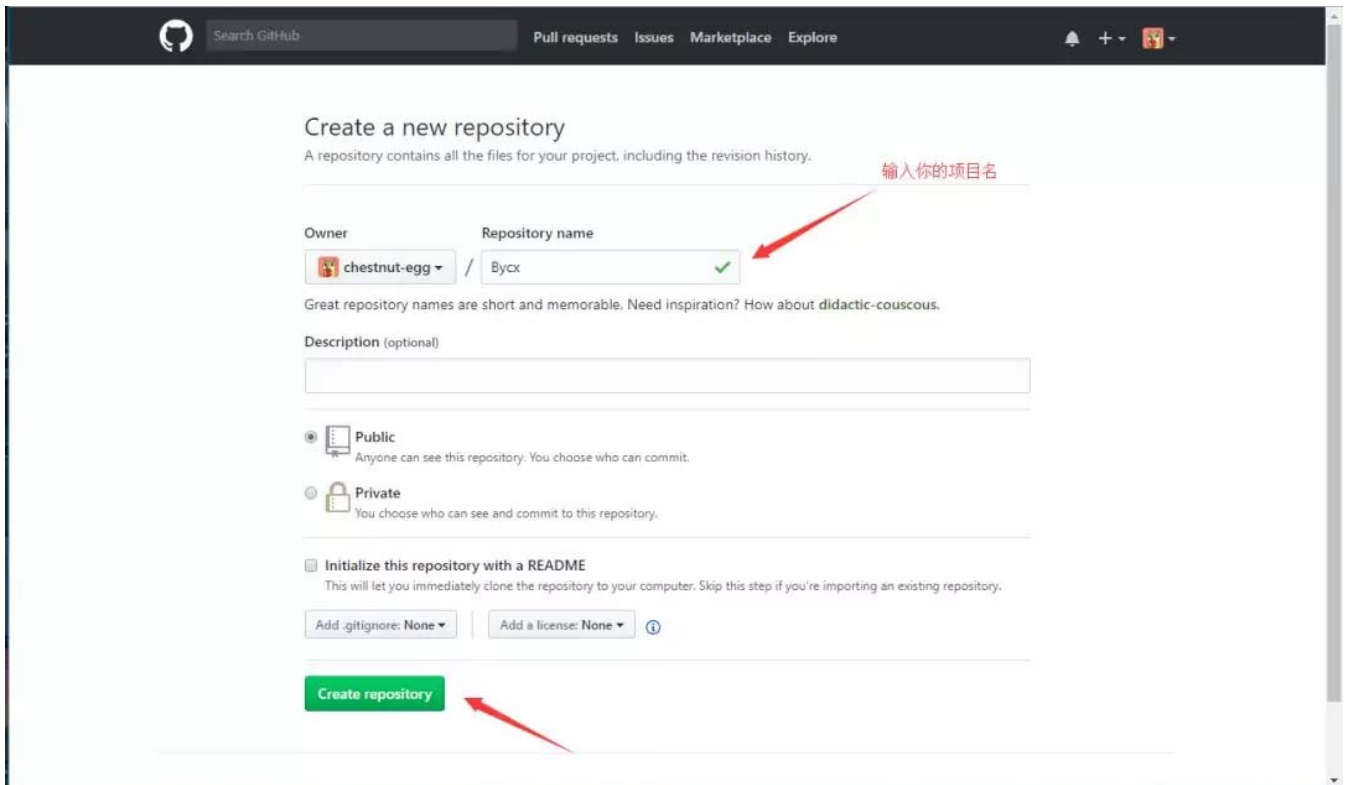


https://github.com/chestnut-egg

3.创建项目







#### 4.回到之前的本地仓库文件夹

执行之前说过的命令

```
git add .
git commit -m "你的备注"
```

然后下面这条使用之前让你记住的地址

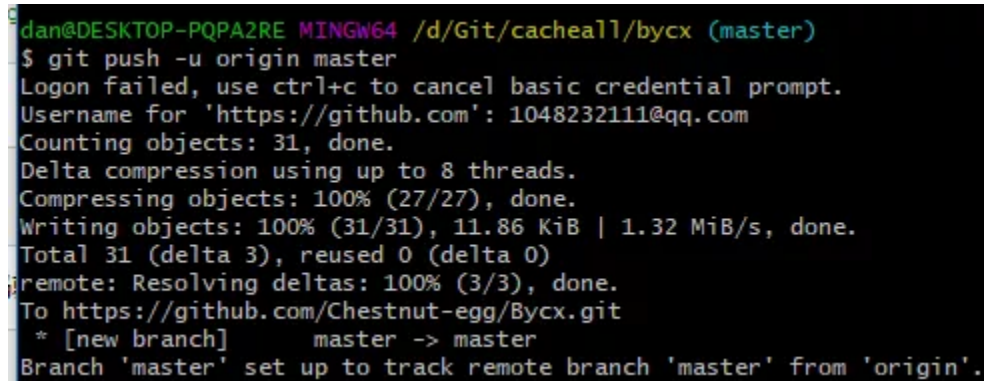
```
git remote add origin https://github.com/....
```

如果是第一次就执行

```
git push -u origin master
```

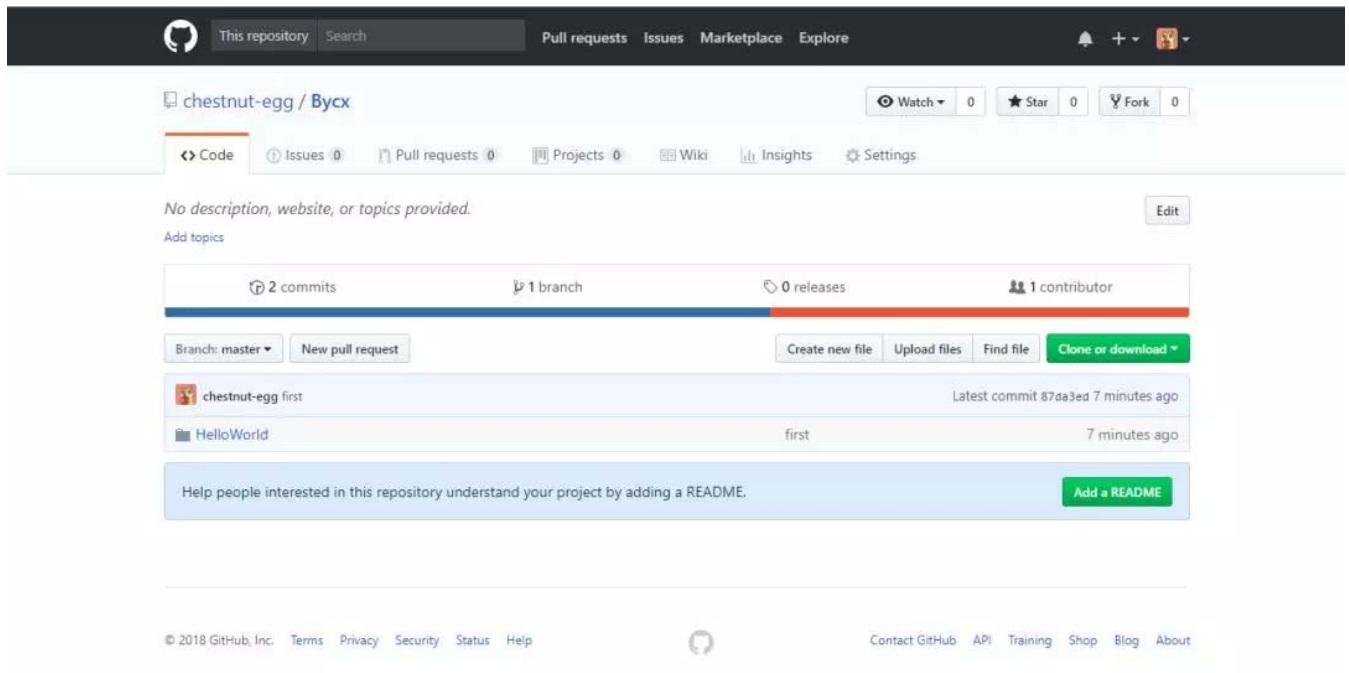
如果不是第一次，是上传更新后的代码就执行

```
git push origin master
```

A terminal window screenshot showing the output of a git push command. The prompt is 'dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)'. The command entered is '\$ git push -u origin master'. The output shows a login failure for 'https://github.com' with the username '1048232111@qq.com'. It then shows the progress of pushing 31 objects, including delta compression and writing. The final output indicates that the remote branch 'master' is set up to track the local 'master' branch from 'origin'.

```
dan@DESKTOP-PQPA2RE MINGW64 /d/Git/cacheall/bycx (master)
$ git push -u origin master
Logon failed, use ctrl+c to cancel basic credential prompt.
Username for 'https://github.com': 1048232111@qq.com
Counting objects: 31, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (27/27), done.
Writing objects: 100% (31/31), 11.86 KiB | 1.32 MiB/s, done.
Total 31 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/Chestnut-egg/Bycx.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

5.在GitHub上可以看到项目文件和提交记录了



最后我把命令都汇总了一下，方便使用

```
git config --global user.name ""
git config --global user.email ""

git init

git add .
git status
git commit -m ""

git remote add origin https://github.com/....

git push -u origin master
git push origin master
```



### Python爱好者社区历史文章大合集：

Python爱好者社区历史文章列表（每周append更新一次）



福利：文末扫码立刻关注公众号，“**Python爱好者社区**”，开始学习Python课程

关注后在公众号内回复“**课程**”即可获取：

**小编的Python入门免费视频课程！！！！**

**【最新免费微课】小编的Python快速上手matplotlib可视化库！！！！**

崔老师**爬虫实战案例**免费学习视频。

陈老师[数据分析报告制作](#)免费学习视频。

[玩转大数据分析！Spark2.X+Python 精华实战课程](#)免费学习视频。

