1. git是一个分布式版本控制（工具）系统；github是一个用git做版本控制的项目托管平台
2. 基本概念

仓库（Repository）：存放项目代码，一个项目对应一个仓库，多个开源项目有多个仓库。

收藏（Star）：Star收藏项目，方便下次查看

复制克隆项目（Fork）：点击别人仓库的Fork按键可以自己复制一个同样的仓库（项目），这个仓库是独立存在的。

发起请求（Pull request）：Fork过来的仓库做了修改，可以向原来的仓库发送一个合并请求（Pull request）。

关注（Watch）：关注一个项目，当项目有更新可以收到通知。

事务卡片（issue）：发现被人的代码bug可以发出一个issue，作者看到可以修改。

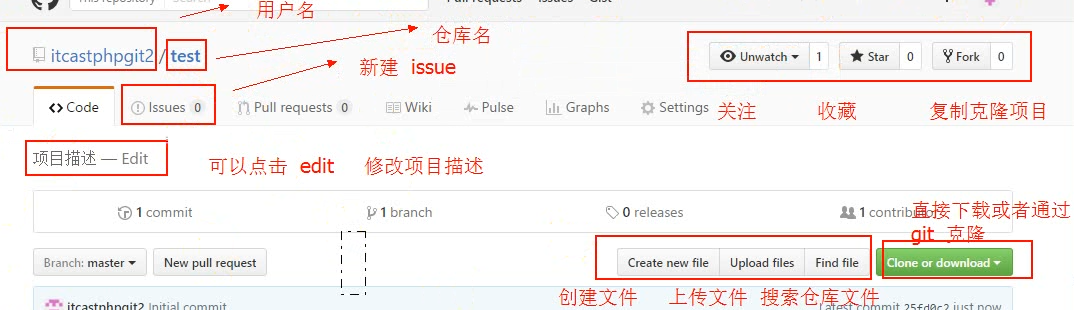
Github主页：显示用户动态，关注的用户或仓库动态，右边显示所有git库

仓库主页：显示项目信息，项目代码，版本，收藏、关注、Fork等

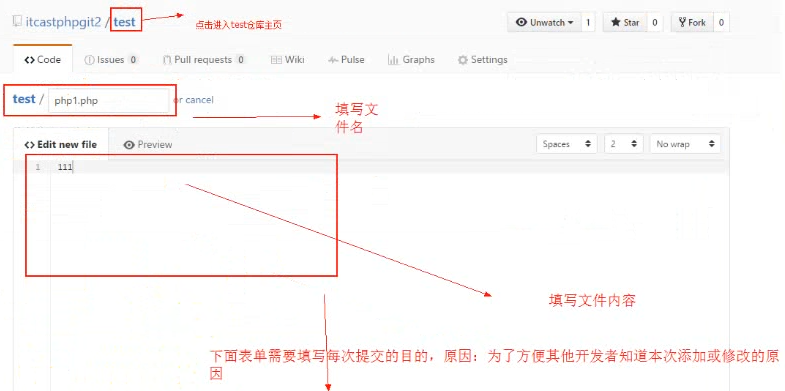
个人主页：个人信息，点击个人根目录或Your Profile进入

1. 一个仓库（git库）一个开源项目通过git管理仓库

仓库主页如下：新建仓库或点击仓库名称可以进入



创建文件，点击Create new file





完成后点击Commit new file提交文件，自动跳转到仓库主页。

可以修改和删除文件，可以在commit里查看提交记录包括删除信息

点击Upload files可以上传文件。

点击Find File或按快捷键“T”可以查找文件。

点击Clone or download可以下载项目

issue作用：打开别人的仓库，发现代码有BUG，但没有成型代码，需要讨论时用；或者使用开源项目出现问题时使用。自己创建的项目有issue选项卡，点击后可以看到别人发来的issue信息列表。Owner和发起issue者可以关闭该issue，存放在closed下。

自己创建的项目选项卡如下：



Fork别人的项目选项卡如下：



1. Pull request流程

fork项目；

修改自己仓库的项目代码；

新建Pull request（按New pull reques）；

等待作者审核后按Merge pull request，合并到原仓库

1. 使用git目的是通过git管理github托管项目代码
2. Git工作区域，新建修改代码；暂存区，暂存已经修改的文件，最后统一提交到git仓库；git 仓库最终确定的文件保存到仓库

通过git add file把文件提交到暂存区，通过 git commit –m ‘本次提交描述’提交到仓库，通过git status可以查看状态

1. Git基本信息设置

设置用户名：

git config --global user.name 你的目标用户名

设置邮箱：

git config --global user.email 你的目标邮箱名

查看设置：

git config --list

1. 初始化一个新的git仓库

a.创建文件夹

b.在创建的文件夹下初始化git仓库: git init，在新建的文件夹下生成.git文件夹存储仓库所有信息。

c.向新的仓库添加文件:创建文件；向暂存区添加文件git add 文件；提交暂存区文件到仓库git commit –m ‘本次提交描述’

d.修改仓库文件：修改文件；向暂存区添加文件git add 文件；提交暂存区文件到仓库git commit –m ‘本次提交描述’

d.删除仓库文件：修改文件；从git中删除文件git rm 文件；提交暂存区文件到仓库git commit –m ‘本次提交描述’

1. Git远程仓库，作用：备份，代码共享

本地创建ssh key：cd ~ #保证当前路径在”~”下

ssh-keygen -t rsa -C "your\_email@youremail.com自己的邮箱"

在github setting里Add SSH key把生成的SSH KEY(在id\_rsa.pub中)复制进去

验证是否成功：ssh -T [git@github.com](mailto:git@github.com)

先将远程仓库（github对应项目）复制到本地：git clone <https://github.com/XXX/XXX.git>

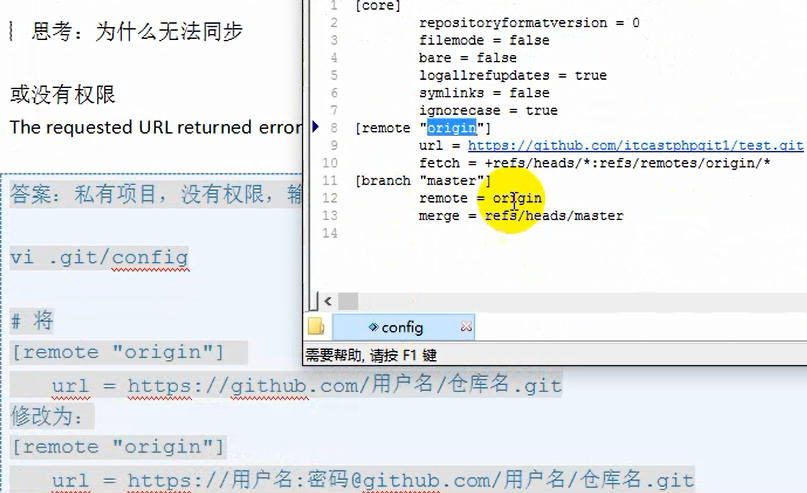
在本地仓库做操作，在推送（同步 ）到远程仓库

命令：git push

Git 2.x 中git push时遇到 push.default 警告的解决方法：

git config --global push.default simple #或输入git config --global push.default matching

其它问题



七步大法教你将新项目上传到GitHub

在一切开始之前，你需要打开Git Bash（Git命令行工具），并使用cd命令进入要上传的新项目的根目录下。

**第一步：**

添加 README.md 文件和 .gitignore文件.gitignore 的作用是告诉Git哪些文件不需要添加到版本管理中，可参考https://github.com/github/gitignore提供的非常全的 gitignore模板 ，按需复制粘贴到刚创建的 .gitignore 文件中，非常方便。

touch README.md;

touch .gitignore;

**第二步：**

建立Git仓库。在当前项目目录中生成一个本地的Git仓库。会生成一个名字为.git的隐藏目录。

git init

**第三步：**

添加当前目录中的所有文件到索引中。

git add .

**第四步：**

将上一步中添加的文件提交到本地仓库中，并附带一个 -m 参数后面加上提交的一段注释，用于说明本次提交的代码都做了什么。

git commit -m "first commit"

**第五步：**

在GitHub上新建一个Repository（远程仓库），假如新建的仓库地址为 https://github.com/stevewei95/example.git，执行以下命令将开始我们建立的本地的仓库关联到这个Github远程仓库上。别名为origin，默认为master分支。

git remote add origin https://github.com/stevewei95/example.git

**第六步：**

在上传到Github仓库之前，我们养成先pull一下的习惯，保证我们拿到的是远程仓库的最新代码，执行以下命令。

git pull origin master

**第七步：**

最后我们把本地仓库中的代码上传到GitHub远程目标仓库中。过程中可能可能会提示输入 Username 和 Password，此时输入GitHub的账号和密码即可。

git push -u origin master

*github创建仓库后提示：*

*…or create a new repository on the command line*

*echo "# tttt1" >> README.md*

*git init*

*git add README.md*

*git commit -m "first commit"*

*git remote add origin https://github.com/am3359/tttt1.git*

*git push -u origin master*

*…or push an existing repository from the command line*

*git remote add origin https://github.com/am3359/tttt1.git*

*git push -u origin master*

1. 个人网站访问：https://用户名.github.io
2. 个人网站搭建步骤：
3. 创建个人站点->新建仓库（仓库名是【用户名.github.io】）
4. 在仓库下新建index.html文件，只能是静态网页

项目站点：<https://用户名.github.io/>仓库名

1. 进入项目（仓库）主页，点击setting选项卡
2. 在setting页面，点击Launch automatic page generator来自动生成主题页面

-------------------------------------------------------------------------------------------------------------------------------

1. git 放弃本地修改 强制更新

git fetch --all

git reset --hard origin/master

1. 以以以
2. 以以以
3. 以以以
4. 以以以