Git和Github的基本用法

背景

git是一个版本控制工具. 主要解决三个问题

- 1. 代码被喵星人吃掉了.
- 2. 产品经理反复修改需求, 需要同时维护多个版本代码.
- 3. 多人协同开发.

Github 是 "全球最大的同性社交网站". 通过 git 可以把代码上传到 Github 上给全球的用户分享.

下载安装

安装 git for windows

这个是一个git的windows系统的命令行版本

https://git-scm.com/downloads

或者

https://pan.baidu.com/s/1kU5OCOB#list/path=%2Fpub%2Fgit

安装 tortoise git

这个是git的图形界面.

https://tortoisegit.org/download/

注意:

- 1. 先安装 git for windows, 再安装 tortoise git
- 2. 安装 git for windows 一路 next 即可.
- 3. 安装 tortoise git 中需要配置 git.exe, 这个是 git for windows 包含的部分. 如果 git for windows 安装成功, 这一步使用默认结果即可.
- 4. 安装 tortoise git 还需要配置姓名和邮箱, 这个尽量和 Github 的邮箱填成一致.
- 5. 两个工具安装完毕后,需要重启电脑才能正确使用.

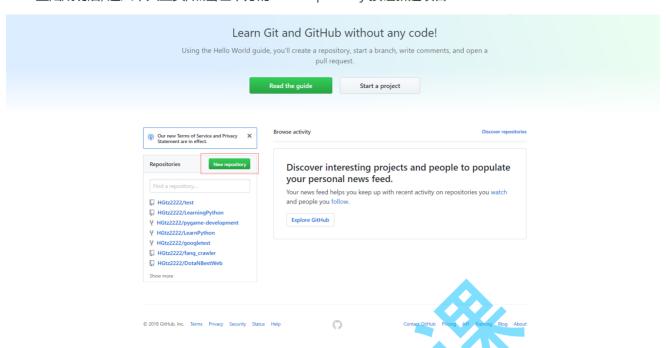
使用 Github 创建项目

注册账号

这个比较简单,参考着官网提示即可.需要进行邮箱校验.

创建项目

1. 登陆成功后, 进入个人主页, 点击左下方的 New repository 按钮新建项目



2. 然后跳转到的新页面中输入项目名称(注意, 名称不能重复, 系统会自动校验. 校验过程可能会花费几秒钟). 校验 完毕后, 点击下方的 Create repository 按钮确认创建.

Create a new repository A repository contains all the files for your project, including the revision history. Owner Repository name HGtz2222 Great repository names are short and memorable. Need inspiration? How about super-duper-eureka. 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 a license: None Add a license: None Create repository

3. 在创建好的项目页面中复制项目的链接, 以备接下来进行下载.

if you've done this kind of thing before or HTTPS SSH https://github.com/HGtz2222/test2.git repository include a README, LICENSE, and .gitignore.

下载项目到本地

- 1. 复制刚才创建好的项目的链接.
- 2. 打开指定的需要放置项目的目录
- 3. 右击目录, 点击 Git Clone
- 4. 在弹出的对话框中输入刚才复制的项目链接即可.



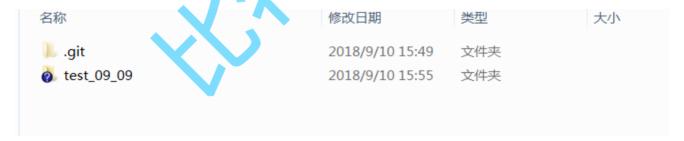
下载成功, 会出现 绿色 图标.

Git 操作的三板斧

放入代码

使用 VS 创建工程,并把工程放在刚才下载到本地的项目路径中.

或者将曾经写过的代码的工程目录直接拷贝到项目目录中.

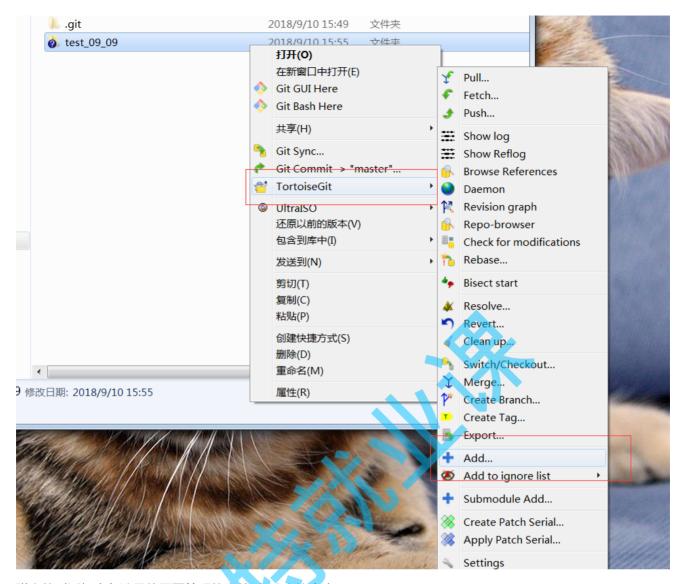


三板斧第一招: git add

告知 git 工具哪些文件需要进行版本管理

此时右击标记为 蓝色?(表示该文件未使用 git 管理)的目录,选择 add

比特就业课-专注IT大学生就业的精品课程



弹出的对话框中勾选具体需要管理的文件. 勾选完毕点击 ok 即可.

此时图标变成红色感叹号(表示该文件被git管理, 但是未提交内容)



三板斧第二招: git commit

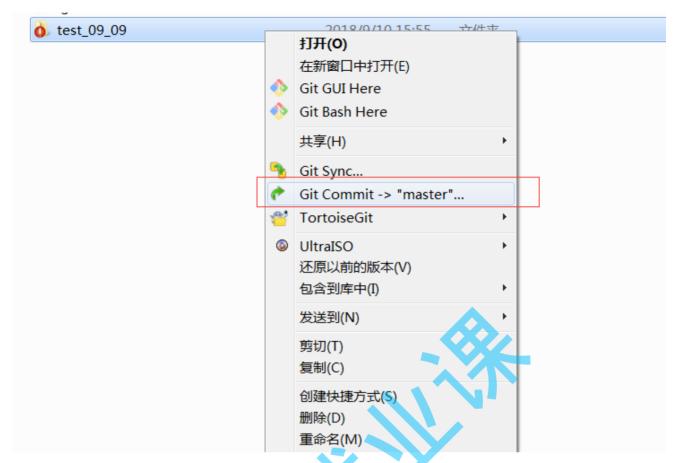
将修改内容提交到本地

每提交一次,就是一个版本.比如开发完某个功能模块,就可以提交一次了.后续进行版本回退都是以提交为准.

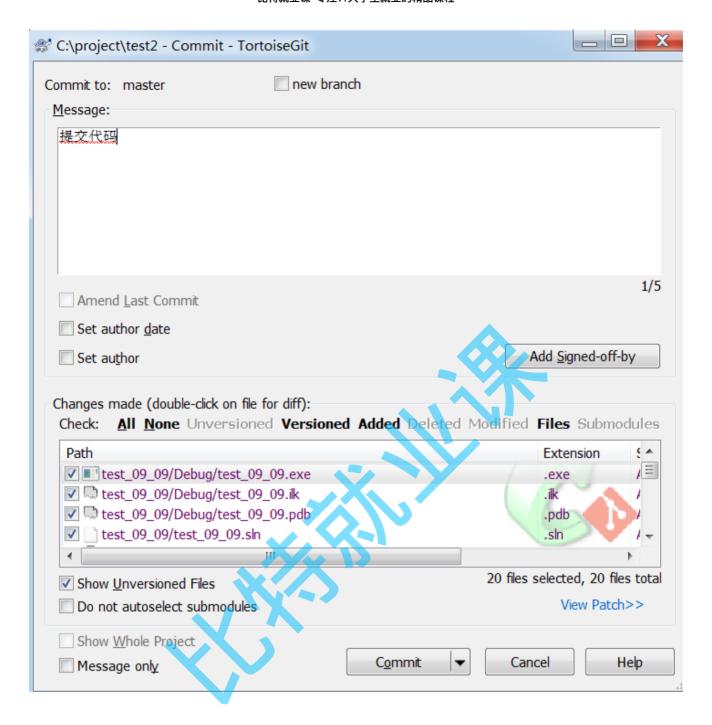
注意: 此时只是提交到本地, Github 上还看不到代码变更.

右键选择 红色感叹号 目录, 选择 Git commit -> master

比特就业课-专注IT大学生就业的精品课程



此时弹出了一个对话框. 可以在此处看到都需要提交哪些文件, 以及每个文件的具体改动情况. 并且需要输入提交日志. 描述这次提交的具体改动原因是什么. 这个日志是后续进行版本回退的重要参考依据.



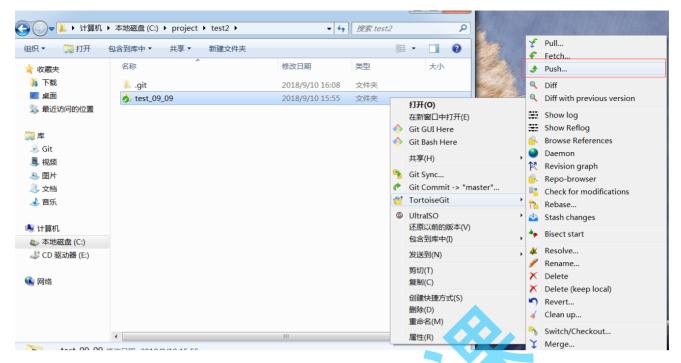
点击下方的 Commit 按钮完成提交.

三板斧第三招: git push

提交的内容需要同步到服务器上,才能让其他人看到改动. 使用 push 即可.

右键需要 push 的目录, 点击 push

比特就业课-专注IT大学生就业的精品课程



弹出的对话框确认 push. 不需要修改, 直接确认即可.

然后会弹出对话框提示输入 Github 的账户和密码.



用户名密码输入正确, 点击 Login, 即可完成 push. 此时刷新 Github 的界面, 就能看到新版本的代码了.

小结

掌握了以上 Git & Github 基本操作, 希望同学们都能够把 Github 用起来, 记录自己的学习过程, 打造自己的专属名片. 如果自己的 Github 日历能够一片绿色, 这是向面试官证明自己靠谱的最有力的证据.

