1.在本机创建一个git版本控制仓库

我们以后项目的代码就在这个目录下，我们每一次开发的版本和历史记录都可以保存在当前创建的git创库中

在电脑的某一个文件夹中

1. 右键->git bash here->在窗口中执行git init
2. 在windows dos窗口中执行，还是执行相同的命令

执行完成后会在当前电脑执行文件夹中生成一个.git文件，证明当前文件夹就是一个本机独立的git版本控制仓库

1. 把工作区中的内容提交到暂存区

暂存区是临时存储，不会生成版本和历史记录的

git add xxx：把执行的文件提交到暂存区

git add . / git add -A：把所有修改的文件都提交到暂存区

Git status 查看当前工作区中问价的状态

红色：还没有提交到暂存区

绿色：还没有提交到历史区

1. 把暂存区的内容同意提交到历史区

提交到历史区的内容，会生成相关的版本和历史记录，以后如果想回滚到某一次的代码，可以使用git命令签出对应的版本即可。

Git commit -m‘xxx’，把暂存区的内容提交到历史区（-m后面的内容都是对本次提交新版本的说明）

中央服务器

真正的项目中，公司一般都会有一台自己内部的中央git仓库管理服务器；我们也可以使用开源免费的git仓库（gitHub）来作为我们的中央服务器仓库；

A可以把自己本地的代码版本信息推送到中央服务器上（其他人也可以）

B如果想看A的信息，只需要在中央服务器上拉取最新的内容即可

GitHub

<https://github.com>

Coding

<https://coding.net/> 中文版的gitHub

1. 基于gitHub创建自己的中央仓库（远程仓库）

--创建完成会生成一个远程仓库地址

HTTPS

<https://github.com/xxx/xxx.git>

SSH

[git@github.com:xxx/xxx.git](mailto:git@github.com:xxx/xxx.git)

1. 把本地的版本代码推送到中央仓库上
2. 让本地仓库和远程仓库建议链接通道

--git remote add [name:一般叫做origin] [远程仓库地址]

--git remote rm[name]

--git remote update [name]

--git remote -v : 查看当前本地仓库所有的链接通道

1. 把本机仓库代码和版本信息推送（拉取）到远程仓库上

--git push origin master 推送

--git pull origin master 拉取

cd /返回根目录，cd../返回上级目录