# 代码管理工具

## Git安装

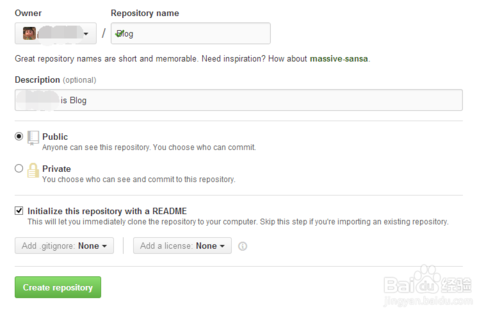
0.TortoiseGit-2.3中文版与**[Git](http://lib.csdn.net/base/git" \o "Git知识库" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank)**安装包: [http://download.csdn.net/detail/renfufei/9738152](http://download.csdn.net/detail/renfufei/9738152" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank)

1. 首先打开 Git 的官方网站: [http://git-scm.com/](http://git-scm.com/" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank)
2. 然后找到下载页面: [http://git-scm.com/downloads](http://git-scm.com/downloads" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank)
3. 找到Windows版本的下载页面: [http://git-scm.com/download/win](http://git-scm.com/download/win" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank)
4. 因为准备使用TortoiseGit做图形客户端,所以就不选择Git GUI 版本. 打开Windows的下载页面后,应该会自动弹出下载框，选择保存即可. 如果没有弹出,可以右键点击页面中的 [click here to download manually.](https://github.com/git-for-windows/git/releases/download/v2.11.0.windows.3/Git-2.11.0.3-64-bit.exe" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank), 选择另存为即可,如果资源下载很慢,可以使用迅雷。
5. 建议将下载好的软件安装包统一分类放到某个目录,如: E:\SOFT\_N\Tools\_ALL\Git\_ALL, 方便以后查找使用.
6. 文件通过浏览器下载完成后,需要修改文件的锁定属性,特别是 **.zip** 文件和 **.chm** 文件(否则打开chm会显示404). 右键点击下载的文件,选择属性,然后点击"解除锁定"按钮,确定即可. 如下图所示
7. 7.可以在cmd里面[测试](http://lib.csdn.net/base/softwaretest" \o "软件测试知识库" \t "http://blog.csdn.net/renfufei/article/details/41647875/_blank)是否设置了Path,是否安装成功. 在CMD中输入 git 或者 git --version 试试:
8. 8.如果按照前面的步骤安装下来,那么 git 程序所在的路径已经添加到系统 PATH 中(path就相当于系统自动查找路径列表),所以可以直接在任意路径的 cmd 下执行 git 命令. 如果没有添加,则需要 cd 切换到Git所在的 bin 目录下,才能执行 git 命令.
9. 9.设置你自己的昵称与email
10. 设置本地机器默认commit的昵称与Email. 请使用有意义的名字与email.
11. git config --global user.name "wangjie"
12. git config --global user.email "wangjie\_rj@163.com"
13. git config --global push.default simple
14. 如果要使用git进行推送,则必须配置 push.default ,否则推送失败. 姓名与Email只用于日志标识.实际推送到GitHub等在线仓库时,要用有操作权限的账号登录.
15. 查看git配置可以使用 -l 参数(l 就是 list 的首字母,L的小写):
16. git config -l
17. 在某个项目根路径下面可以设置单独的Email与姓名.
18. git config user.name "tiemaocsdn"
19. git config user.email "tiemaocsdn@qq.com"
20. 可以看到, 配置单个项目时,少了 --global 参数.
21. 然后到git安装目录上面 C:\Program Files\Git
22. 点击 git-bash
23. 生成ssh-key
24. 将pubkey粘贴到github上面
25. 安装toristgit

## 向github上上传

在github创建仓库用于存储管理本地文件，示例：Blog。

点击账号前的加号（Create New）--New repository，根据引导创建一个Blog仓库。

[](http://jingyan.baidu.com/album/27fa732683ebf546f8271f2e.html?picindex=3" \t "https://jingyan.baidu.com/article/_self)

在本地建一个目录，进去

然后右键选择 git clone (将github文件的目录放进去)

然后右键git commit master

Push

Ok

## Git深入学习

### Git 误git add . 撤回

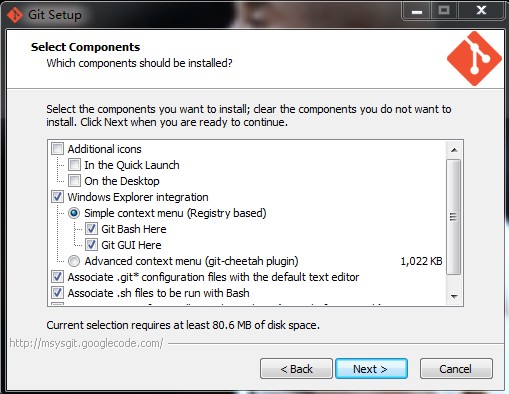
git reset Head filename

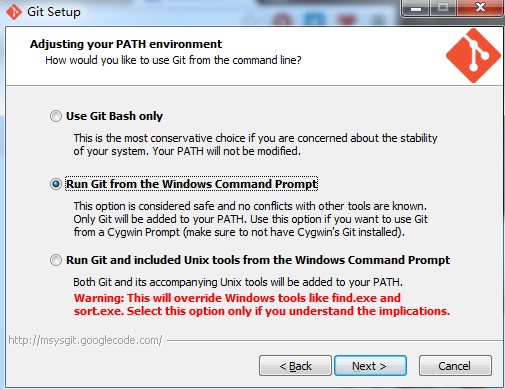
### [Git for Windows安装和基本设置](http://www.cnblogs.com/vitah/p/3612473.html)

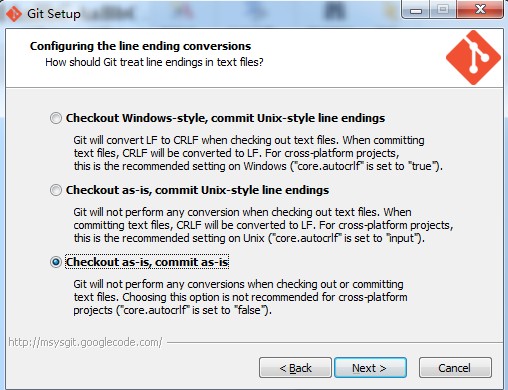
1、下载地址：

<http://msysgit.github.io/>

2、下载完成后安装，安装路径自己选择，其他的选项参照下图：







其他的一步一步往下即可，最后Finish完成安装；

3、配置github的ssh密钥:

(1)打开Git Bash查看电脑上是否已经存在SSH密钥：

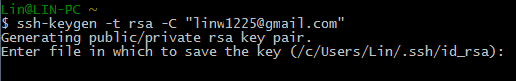
输入 cd ~/.ssh

http://images.cnitblog.com/i/335529/201405/122329362033539.png

若如上图显示无法找到该文件则要创建新的ssh key;

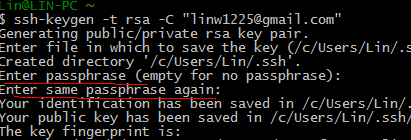
(2)创建新的ssh key:

输入 ssh-keygen -t rsa -C "your\_email@youremail.com"



执行这条命令会如上图提示文件保存路径，可以直接按Enter，

然后提示输入 passphrase（密码），输入两次（可以不输直接两次Enter），



然后会在 .ssh 目录生产两个文件：id\_rsa和id\_rsa.pub

用记事本打开.ssh目录下的id\_rsa.pub文件，复制里面的内容；

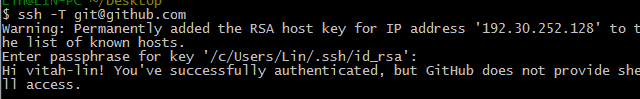
4、复制ssh key到github：

On the GitHub site Click “Account Settings” > Click “SSH Keys” > Click “Add SSH key”

打开github网站，点击右上角扳手图标，然后点击左边菜单的 ssh key， 然后右边页面的 add ssh key，将复制的内容粘贴到github的key中，title可以不填，直接保存即可。

5、测试 ssh 链接 github：

输入 ssh -T git@github.com



出现Successfully就OK；

6、设置自己的git信息：

输入

git config --global user.name "Firstname Lastname" （此处name可修改也不是用于登录github的登录名）

git config --global user.email "your\_email@youremail.com"

设置自己的git信息即完成安装和设置，可以输入git config --list查看自己的git信息。

然后到workspace

右键Git bash here

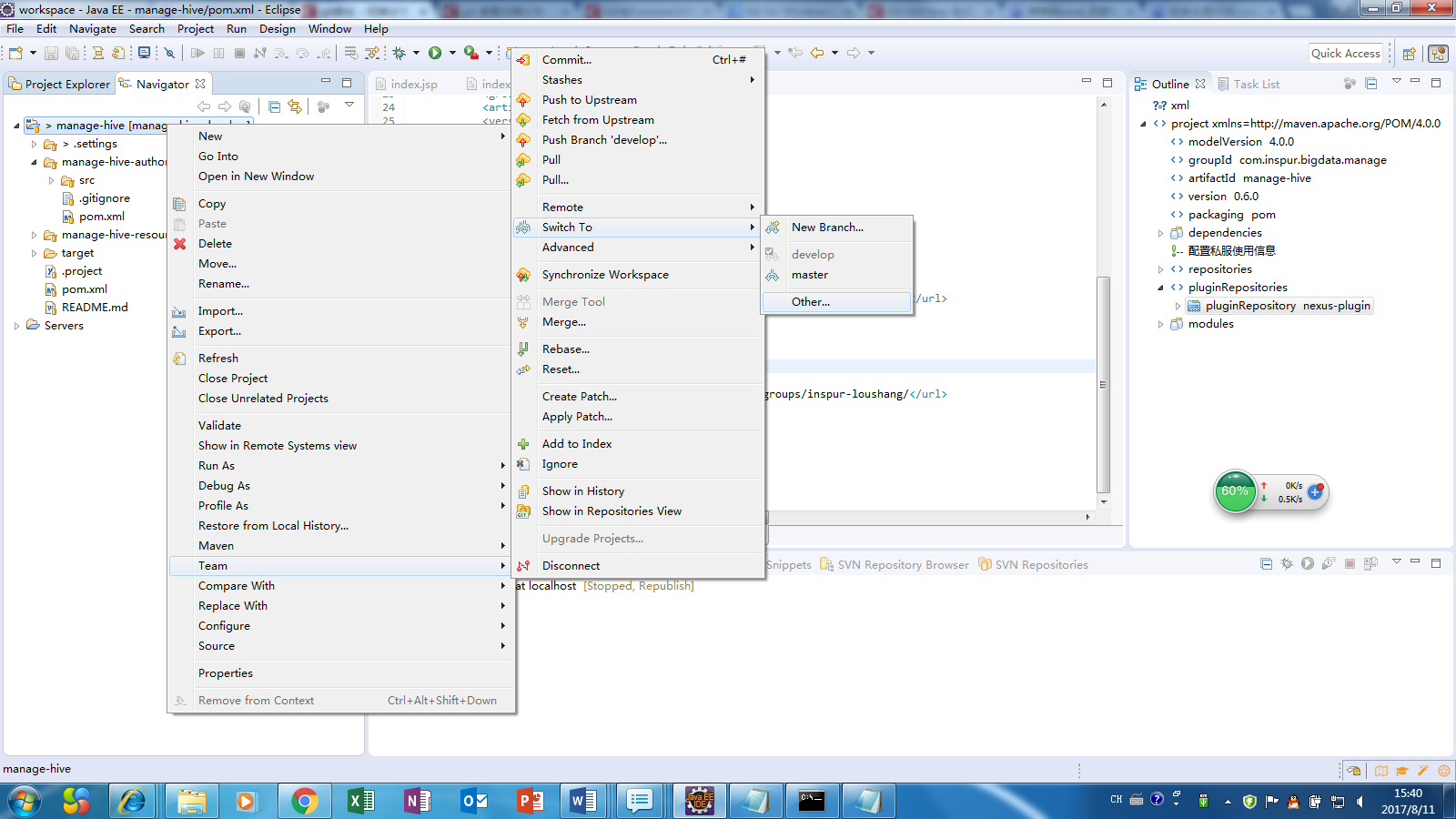
git clone 文件地址

git branch 查看分支

＃切换回master分支

$ git checkout master

或者直接用eclipse的team switch branch:



### 冲突解决

#### 如何丢弃不需要提交的操作

1.场景1：当你改乱了工作区某个文件的内容，想直接丢弃工作区的修改时，用命令Git checkout -- file。

2.场景2：当你不但改乱了工作区某个文件的内容，还添加到了暂存区时，想丢弃修改，分两步，第一步用命令git reset HEAD filename（从暂存区中移除文件），就回到了场景1，第二步按场景1操作。

3.场景3：已经提交了不合适的修改到版本库时，想要撤销本次提交，参考版本回退一节，不过前提是没有推送到远程库。

HEAD指向的版本就是当前版本，因此，Git允许我们在版本的历史之间穿梭，使用命令git reset --hard commit\_id。

穿梭前，用git log可以查看提交历史，以便确定要回退到哪个版本。

要重返未来，用git reflog查看命令历史，以便确定要回到未来的哪个版本。

4.

另一种情况是删错了，因为版本库里还有呢，所以可以很轻松地把误删的文件恢复到最新版本：

$ git checkout -- test.txt

git checkout其实是用版本库里的版本替换工作区的版本，无论工作区是修改还是删除，都可以“一键还原”。

命令git rm用于删除一个文件。如果一个文件已经被提交到版本库，那么你永远不用担心误删，但是要小心，你只能恢复文件到最新版本，你会丢失最近一次提交后你修改的内容。

#### 覆盖更新

git fetch --all

git reset --hard origin/master

git fetch 只是下载远程的库的内容，不做任何的合并 git reset 把HEAD指向刚刚下载的最新的版本

适用的情况是，你修改了代码，使用完了之后，向回退到之前的代码。

#### 配置项导致冲突

如果系统中有一些配置文件在服务器上做了配置修改,然后后续开发又新添加一些配置项的时候,

在发布这个配置文件的时候,会发生代码冲突:

error: Your local changes to the following files would be overwritten by merge:  
        protected/config/main.php  
Please, commit your changes or stash them before you can merge.

如果希望保留生产服务器上所做的改动,仅仅并入新配置项, 处理方法如下:

git stash

git pull

git stash pop

然后可以使用git diff -w +文件名 来确认代码自动合并的情况.

反过来,如果希望用代码库中的文件完全覆盖本地工作版本. 方法如下:

git reset --hard

git pull

其中git reset