**一、安装**

默认安装：[msysGit](http://code.google.com/p/msysgit/downloads/detail?name=Git-1.7.0.2-preview20100309.exe)

**二、配置**

1、C:\Program Files\Git\etc\gitconfig 添加：  
**【注意！】请将第二行最后的 “your-id” 修改成你在服务器上的实际 id，默认是姓名拼音。**

[alias]

go = "! bash -c \"git pull && git add .; if [ \\\"$\*\\\" == \\\"\\\" ]; then git commit -a; else git commit -am \\\"$\*\\\"; fi; git push origin master:your-id;\""

[core]

autocrlf = false

[gui]

encoding = utf-8

[i18n]

commitencoding = GB2312

[user]

email = xxx@gmail.com

name = 某某某

2、C:\Program Files\Git\etc\inputrc 修改两行为：

set output-meta on

set convert-meta off

3、C:\Program Files\Git\etc\git-completion.bash 末尾增加：

alias ls='ls --show-control-chars --color=auto'

4、C:\Program Files\Git\etc\profile 末尾增加：

export LESSCHARSET=utf-8

【TIP】以上文件最好使用支持 unix 格式的编辑器修改（如 Notepad++、NetBeans），最次也用“写字板”而非“记事本”。

【TIP】若想了解为什么这样设置，请参见：[Windows 下 Git 客户端的选择，及 msysGit 各种中文问题的解决](http://wangcongming.info/2010/07/31/windows-%E4%B8%8B-git-%E5%AE%A2%E6%88%B7%E7%AB%AF%E7%9A%84%E9%80%89%E6%8B%A9%EF%BC%8C%E5%8F%8A-msysgit-%E5%90%84%E7%A7%8D%E4%B8%AD%E6%96%87%E9%97%AE%E9%A2%98%E7%9A%84%E8%A7%A3%E5%86%B3/)

**三、生成密钥**

安装完后，需要生成一对 Key（这里指密钥），然后才能通过加密的方式和服务器的代码库取得同步。

到开始菜单，找到“Git Bash”，运行之，并执行以下命令：

$ ssh-keygen -t rsa

程序会提示您输入密钥的文件名，直接按回车即可。  
然后会要求你输入一个密码，将来在使用密钥的时候需要提供这个密码。可以输入，也可以不输入直接回车（无论输入还是不输入，都会要求你确认一次）。  
确认完毕后，程序将生成一对密钥存放在以下文件夹：

C:\Users\Administrator[这里替换成你的用户名]\.ssh

密钥分成两个文件，一个私钥（id\_rsa）、一个公钥（id\_rsa.pub）。  
私钥保存在您的电脑上，公钥交项目负责人添加到服务器上。用户必须拥有与服务器公钥所配对的私钥，才能访问服务器上的代码库。

**【注意！】为了项目代码的安全，请妥善保管你的私钥！因为一旦私钥外泄，将可能导致服务器上的代码被泄漏！**

**四、使用**

**1、克隆代码库**

使用 Windows 资源管理器，打开你打算存放项目代码的文件夹，点右键选择 Git Bash。

在我们的项目管理系统中，每个项目的首页，都有写明代码克隆的地址，比如我们用于测试目的的沙盒项目：

$ git clone your-name@testing.aysaas.com:/var/projects/sandbox

在 Git Bash 中运行这条命令就能将沙盒项目中的所有代码（其实只是几个随便测试的文件）克隆到本地。

接着您就可以打开习惯的 IDE（如 NetBeans），投入到项目的开发中啦～！

【TIP】上面命令中的 your-name 要改成你在服务器上实际的用户名。

**2、查看修改差异**

开发过程中，如果你想了解修改了哪些代码，总览所有代码的改动情况，可以在 Git Bash 中输入此命令：

$ git diff

【TIP】Git Bash diff 的时候有两个缺点：一、窗口太窄，可能显示不下整行的代码；二、如果代码中有中文，会乱码。如果你碰到这两个问题，可以在项目文件夹下点右键，选择 Git Gui。

**3、提交修改**

每当完成一个阶段的代码，就需要提交代码以记录进展，方便日后查找问题以及团队协作。

$ git go aaa 修改说明（改动了什么？为什么这样改？）

【TIP】别忘了 go 后面的 aaa，关于 ‘git go’ 命令的详细说明，请参见 [Windows 下 Git 客户端的选择，及 msysGit 各种中文问题的解决](http://wangcongming.info/2010/07/31/windows-%E4%B8%8B-git-%E5%AE%A2%E6%88%B7%E7%AB%AF%E7%9A%84%E9%80%89%E6%8B%A9%EF%BC%8C%E5%8F%8A-msysgit-%E5%90%84%E7%A7%8D%E4%B8%AD%E6%96%87%E9%97%AE%E9%A2%98%E7%9A%84%E8%A7%A3%E5%86%B3/)

**【TIP】请尽量养成勤提交的好习惯。当代码不幸出现问题时，比较容易找出从什么时刻开始出现问题，并回退到该时刻进行调试，最大限度保护已完成的阶段性工作。**

【TIP】以上命令，都需要在项目目录下运行。Git Bash 在命令提示符前，会显示当前所在的目录。如果当前不在项目目录之下，需要用 cd 命令切换到项目所在目录。  
简单的办法，就是先在资源管理器里打开项目文件夹，再点右键，选择 Git Bash。

**五、总结**

至此，从获取代码、查看差异、到提交代码，整个流程都熟悉了。Git 还有比较高级的技巧，大家可以参考 [Git 进阶功能](http://wangcongming.info/2010/07/31/git-%E7%B3%BB%E5%88%97%E4%B9%8B%E5%9B%9B%EF%BC%9Agit-%E8%BF%9B%E9%98%B6%E5%8A%9F%E8%83%BD/) 或在线找进一步的资料学习。