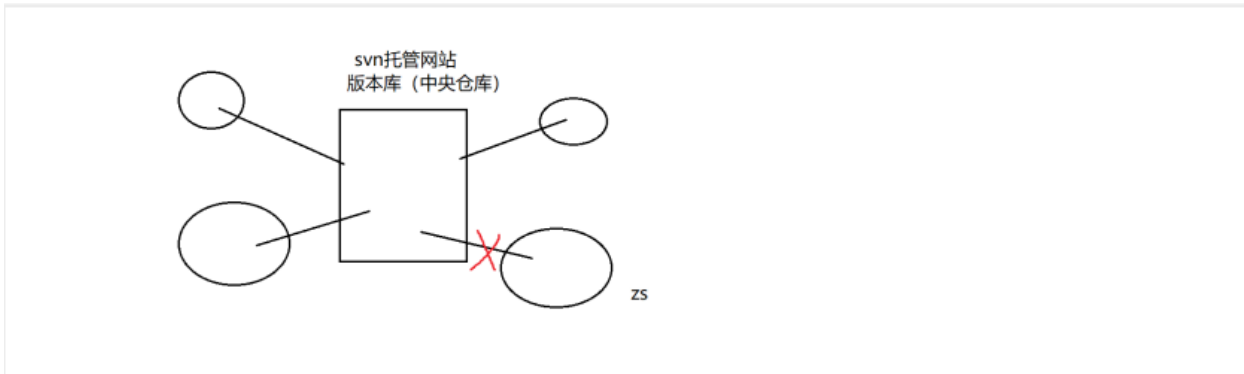
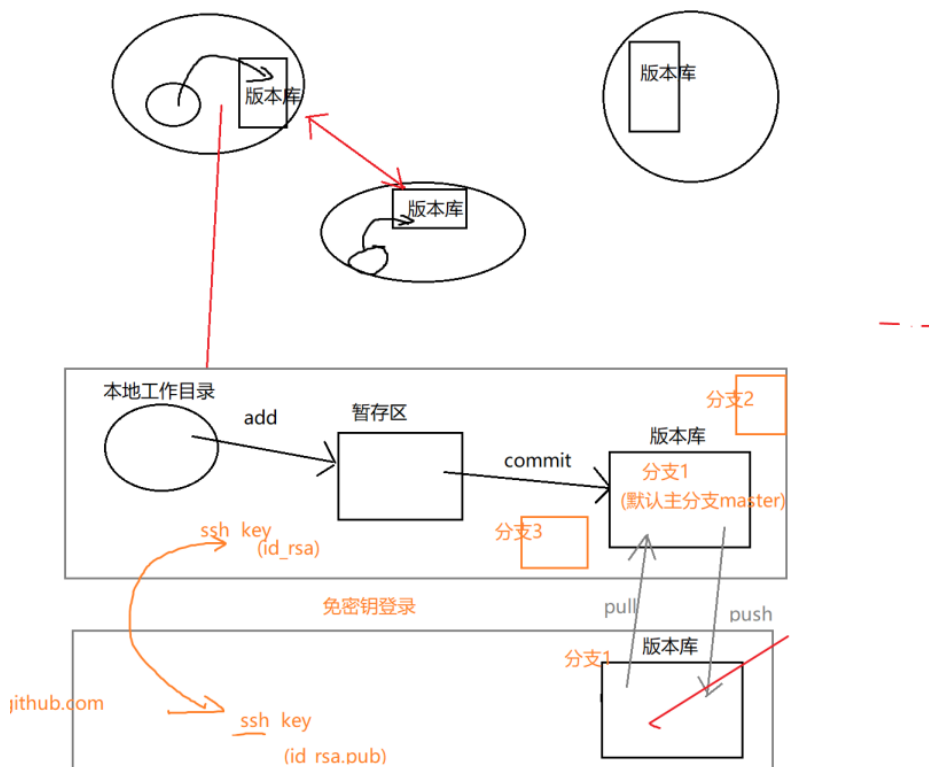


SVN冲突：为了避免冲突,要及时提交、更新



git：分布式版本控制软件



版本库叫分支, 有好几个分支, 默认主分支为master分支, commit默认提交到master分支

命令：

**git add**: 将本地文件 增加到暂存区

**git commit**: 将暂存区的内容 提交到 本地仓库（本地分支，默认master分支）

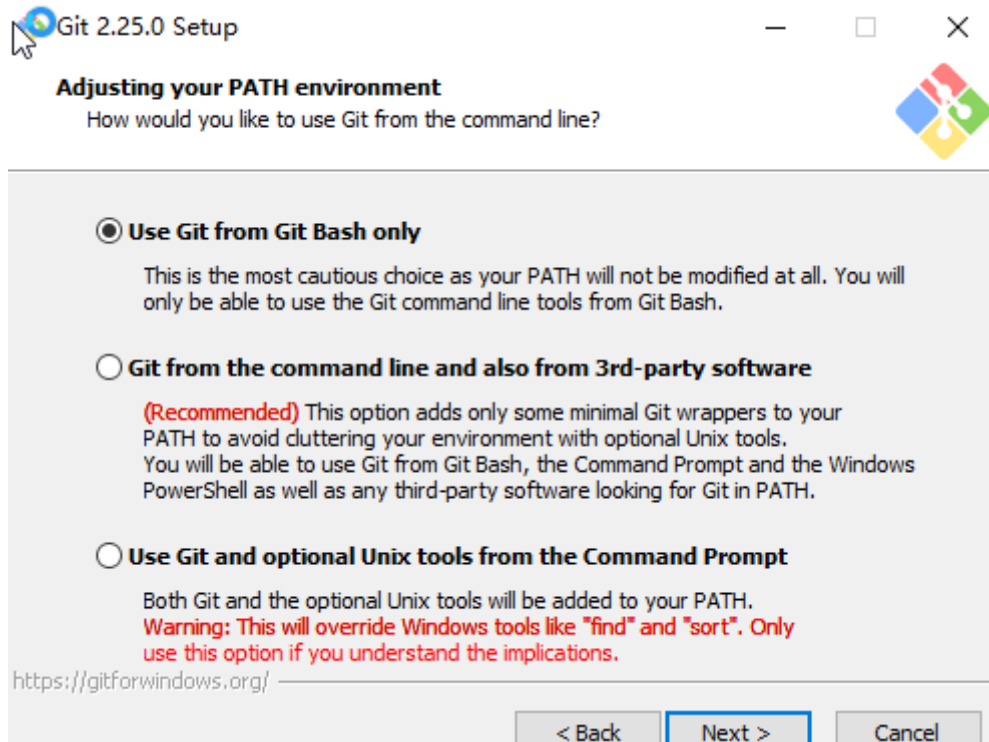
**git push**: 将本地仓库的内容 推送到 远程仓库（远程分支）

`git pull`: 将远程仓库（远程分支）的内容 拉取到 本地仓库（本地分支）

## 1. 安装git

<https://github.com/waylau/git-for-win>

安装时: Use git from git bash only..., 其他默认下一步



## 2. 配置path: E:\programs\Git\bin

## 3. 配置git: 用户名和邮箱

在桌面单击右键 点击-git bash

```
git config --global user.name "jcl"
```

```
git config --global user.email "1345414527@qq.com"
```

4.查看C:\Users\YANQUN\.gitconfig看看是否配置成功 (可能目录不一样)

5.搭建git服务器 (远程仓库) : 统一的托管网站  
(<https://github.com/>)

为了 在本地 和远程仓库之间进行 免密钥登录, 可以配置ssh

ssh:本地-远程

6.配置ssh: 先在本地配置, 发送给远程(本地和远程ssh关联)

a. 现在本地生成ssh:

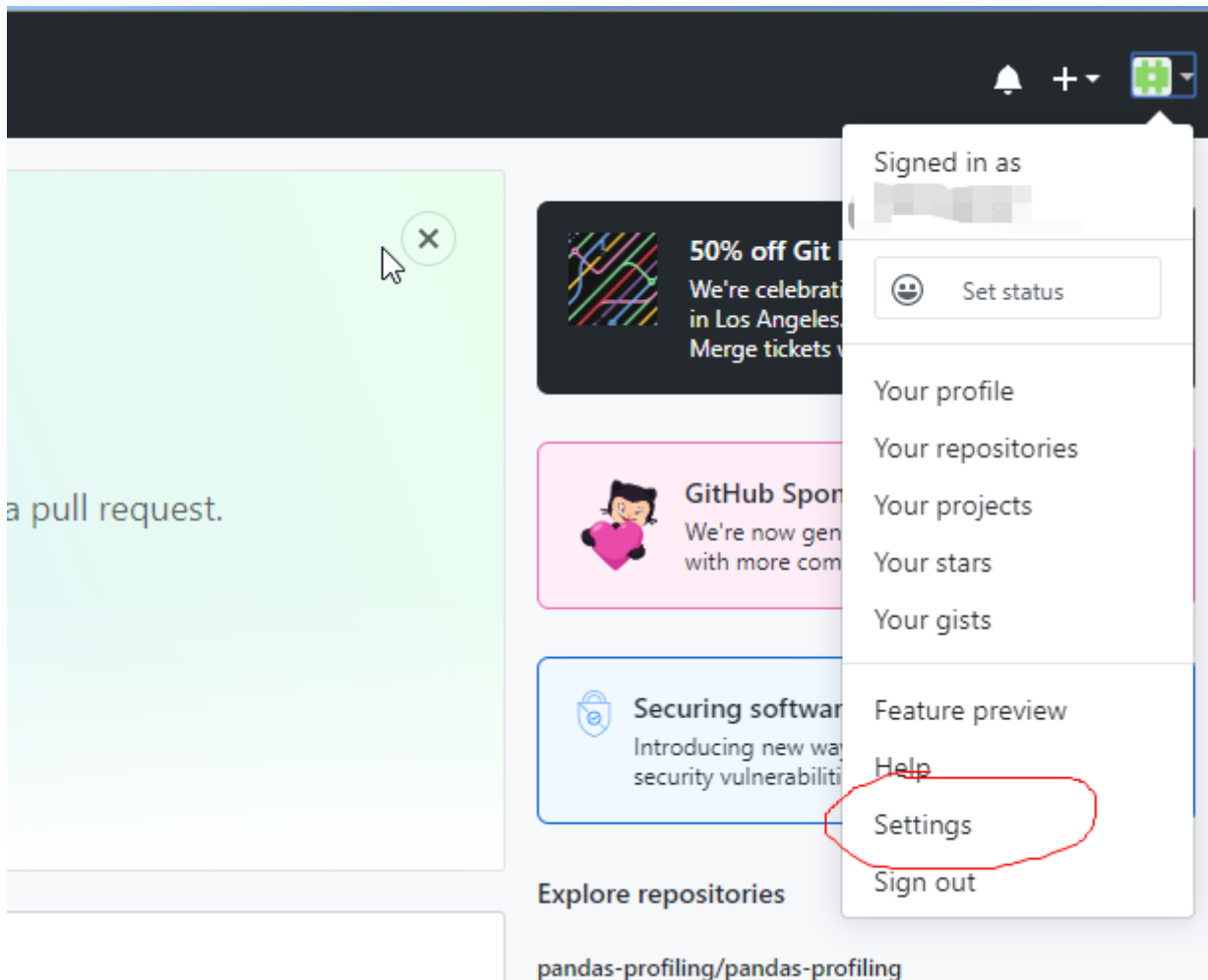
ssh-keygen -t rsa -C 1345414527@qq.com 一直回车

```
$ ssh-keygen -t rsa -C 1345414527@qq.com
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/DELL/.ssh/id_rsa):
Created directory '/c/Users/DELL/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/DELL/.ssh/id_rsa.
Your public key has been saved in /c/Users/DELL/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:zjHl5Omdafd/IxI5WkJQL1Icw/OIzEyKAYd4Fj7B4RY 1345414527@qq.com
The key's randomart image is:
+---[RSA 3072]-----+
|  oE=o..+=. |
|.O=+ oo.+o |
|O= .=.O.*. |
|. . =.B.O |
| S = . |
| o = O o |
| o = O |
|. o o . |
|. o.o |
+-----[SHA256]-----+
```

b. 发送给远程:

进入github - settings - SSH and ... - New SSH - title任意、key  
中输入 刚才在本  
地生成的 ssh:

将本地刚才生成的id\_rsa.pub内容复制到远程的Key中



Personal settings

- File
- Account
- Security
- Security log
- Emails
- Notifications
- Billing
- SSH and GPG keys**
- Blocked users
- Repositories
- Organizations
- Saved replies
- Applications

Developer settings

### SSH keys / Add new

Title

Key

Begins with 'ssh-rsa', 'ssh-ed25519', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', or 'ecdsa-sha2-nistp521'

Add SSH key

## 7.测试连通性:

```
ssh -T git@github.com
```

如果本地和远程成功通信，则可以在 /.ssh目录中 发现known\_hosts文件  
如果失败：多尝试几次 、检查回车符

## 8.本地和远程项目关联

### 1..在本地新建git项目

在项目根目录 右键 - git bash    git init    会出现一个.git文件夹

### 2..在远程建立git项目

new-建立项目- 生成 <https://github.com/1345414527/studyTest.git>

### 3.本地项目-远程项目关联

```
git remote add origin git@github.com:1345414527/studyTest.git
```

## 第一次发布项目 （本地-远程）

```
git add .            //文件-暂存区    .表示所有文件  
git commit -m "注释内容"    //暂存区-本地分支（默认master）  
git commit    //之后再写注释  
git push -u origin master
```

## 第一次下载项目 （远程-本地）

```
git clone git@github.com:yanqun/mygitremote.git
```

## 之后提交(本地-远程)

(在当前工作目录 右键-git bash)

```
git add.  
git commit -m "提交到分支"
```

`git push origin master` //少了个u

## 更新(远程-本地)

`git pull`

Egit: 在Eclipse中操作git

目前的eclipse基本都支持git, 如果不支持 则到eclipse marketplace 搜git安装

配置:

a. team-git-configuration -邮箱 用户名

b. general -network -ssh2选中 生成的ssh目录

第一次发布

`share project`

加入暂存区`add to index`

提交到本地分支`commit`

将项目推送到远程                      右键-team -remote -push ---

提交

`team-add to index`

`team -commit`

`team -push`

commit时:

commit and push 或commit按钮的区别:

commit按钮: 不能单独的Push某一个文件, 只能Push整个项目

commit and push: 可以 单独Push某一个文件

第一次下载

`import -clone` -输入 https/ssh的唯一标示符

更新

`team - remote -pull`

git冲突的解决:

发现冲突:            进入同步视图    右键——`team - synchronized...`

解决:

添加到本地暂存区 `add to index`

提交到本地分支 `commit`

更新服务端的分支内容    到本地分支 `pull`

修改冲突: 直接修改 或者 `merge tool`

(--->已经变为了普通本地文件)

`add to index`

`commit push`

