

Git协作教程及IDEA项目简介

@author 黄雨生

@date 2017.12.3

本文旨在使大家快速掌握使用Git进行协作开发的方法。另外，本文也对如何使用IDEA进行开发及如何通过IDEA运行项目进行了简单的介绍。需要注意的是，windows用户可能需要使用一些shell工具(Powershell之类的)才能执行文中的一些命令。

Git

1. 安装Git和注册GitHub账号

//大家自行解决。Git最好安装命令行版本的，不要安装GUI版本的。

2. 设置Git的用户名和邮箱

通过以下命令设置本地Git的用户名和邮箱：

```
git config --global user.name "这里输入你的Github用户名"  
git config --global user.email "这里输入你的Github邮箱"
```

3. 创建SSH key

在命令行中输入以下命令：

```
ssh-keygen -t rsa -C "这里输入你的Github邮箱"
```

命令完成后可以在主目录下找到.ssh目录，其中包含了id_rsa和id_rsa.pub两个文件，打开id_rsa.pub文件，复制其内容。接着使用浏览器登录你的GitHub账号，在profile中有一个SSH key选项，点击后选择添加一个SSH key，title可以随便取名，然后将复制的内容粘贴进去即可。

4. fork项目

目前原始项目托管在我的GitHub上，通过<https://github.com/ShengAndXian/TAS>可以访问。进入之后，点击页面右上角的fork按钮就可以把项目fork到你自己的GitHub里了。现在，你有了原始项目的一个副本，你可以对这个副本做任何事情而不会影响原始项目。

5. clone项目到本地

进入你自己的项目界面，你会看到一个绿色的按钮“clone or download”，点击按钮会出来一个链接，复制这个链接。之后打开命令行界面，cd到任何一个目录，输入以下命令：

```
git clone 这里粘贴已复制的链接
```

完成之后你会看到目录下生成了一个名为TAS的文件夹。

6. 添加远程仓库

在clone自己的项目副本的时候，我们就已经有了一个远程仓库，在命令行中可以输入以下命令查看：

```
git remote
```

clone时自动添加的远程仓库名称一般为origin。所以输入该命令，你会看到有一个名为origin的远程仓库。

接下来，我们需要添加另一个远程仓库，这个仓库就是原始项目仓库，也就是托管在我的GitHub上的项目。在命令行中输入以下命令：

```
git remote add origin-hys https://github.com/ShengAndXian/TAS.git
```

这样就会将原始项目添加到你的远程仓库中，而这个远程仓库的名称为origin-hys。

7. 创建分支

一般来说，我们不会直接在master分支上进行开发，而是创建一个develop分支用于日常开发。

首先cd到TAS文件夹下，接着输入以下命令：

```
git branch develop //创建develop分支  
git checkout develop //切换到develop分支
```

8. 开发

在develop分支下我们就可以进行自己的开发了，开发过程中你可以不断的commit自己的修改。至于如何commit这里就不详细讲了，可以自行谷歌。

9. merge & push & pull request

假设你已经完成了某个功能或者某个界面的开发，并且所有的修改都已经被commit到develop分支，这时候你应该会想将自己的修改提交到原始项目。那么应该怎么做呢？参考以下一系列命令：

```
git fetch origin-hys //将原始项目的更新抓取到本地。
```

```
git checkout develop //切换到develop分支  
git merge origin-hys/master //将原始项目合并到develop分支
```

```
git checkout master //切换到master分支  
git merge develop //将develop分支合并到master分支
```

```
git push origin //将更新推送到自己的项目副本
```

通过浏览器登录GitHub，进入自己的项目仓库，你会看到项目已经被更新了。然后点击pull request按钮，填写完相应信息后，即可向原始项目发起合并请求。如果通过review，你的更新就会被合并到原始项目中。

note:这里我们没有讨论合并冲突的问题，因为解决起来还是比较复杂的，到时候如果出现了问题我们再讨论。

IDEA

1. 安装IDEA

为了在自己的电脑上运行整个项目，必须下载IDEA。IDEA可以通过官网下载，选择ultimate版本，通过学生认证后即可使用。

2. 打开项目

之前已经通过Git将项目clone到了本地，因此只需要简单的将文件夹拖入IDEA中打开即可。

3. 项目结构

- 整个项目是基于maven构建的，因此你会在根目录下看到一个名为pom.xml的文件，该文件描述了整个项目的全部外部依赖。另外，需要打开maven的自动导入功能。
- 根目录下有一个名为tas.log的文件，该文件是一个日志文件，当我们遇到错误时可以去该文件中查看对应的日志。
- /src/main/java目录下存放我们所有的java源文件，后端开发的同学主要在这个目录下工作。
- /src/main/java/com/se目录下有一个Application类，这是一个主类，用于启动整个项目，因此不要修改它。
- /src/main/resources/static目录下存放所有的静态资源，包括css文件、js文件以及一些图片。静态资源可以直接通过url进行访问。前端的同学可以将css文件、js文件放在这里。
- /src/main/resources目录下的其他文件夹都不能直接通过url进行访问，因此可以用来存放视频、课件等资料。
- /src/main/resources/application.properties文件用于配置spring。
- /src/main/webapp目录是整个web项目的基础。
- /src/main/webapp/WEB-INF目录下有一个web.xml文件，它可以用来配置web。由于采用了spring boot框架，因此基本上不需要手动配置web.xml。
- /src/main/webapp/WEB-INF/view目录下存放所有的jsp文件，前端开发的同学主要在这个目录下工作。

4. 运行项目

首先确保所有的外部依赖都已经安装好了。

点击工具栏中的view —> 选择tool window —> 选择maven project —> 展开plugins —> 展开spring-boot —> 双击spring-boot: run。

启动之后，打开浏览器在地址栏中输入localhost:8080/login即可进入登录界面。

5. 调试

大家在开发的过程中，可以随时运行项目来观察系统的行为是否符合预期，然后根据反馈修改自己的代码。

End.