# 课程介绍

1、张三、李四 a.java

2、重要性

# 课程安排

## Git 版本控制工具

下载、安装、指令、集成idea

## GitHub 远程代码托管库 国外

push 推送 （本地项目推到远程仓库）

pull 拉取

clone

…

集成idea

## Gitee 远程代码托管库 国内

push 推送 （本地项目推到远程仓库）

pull 拉取

clone

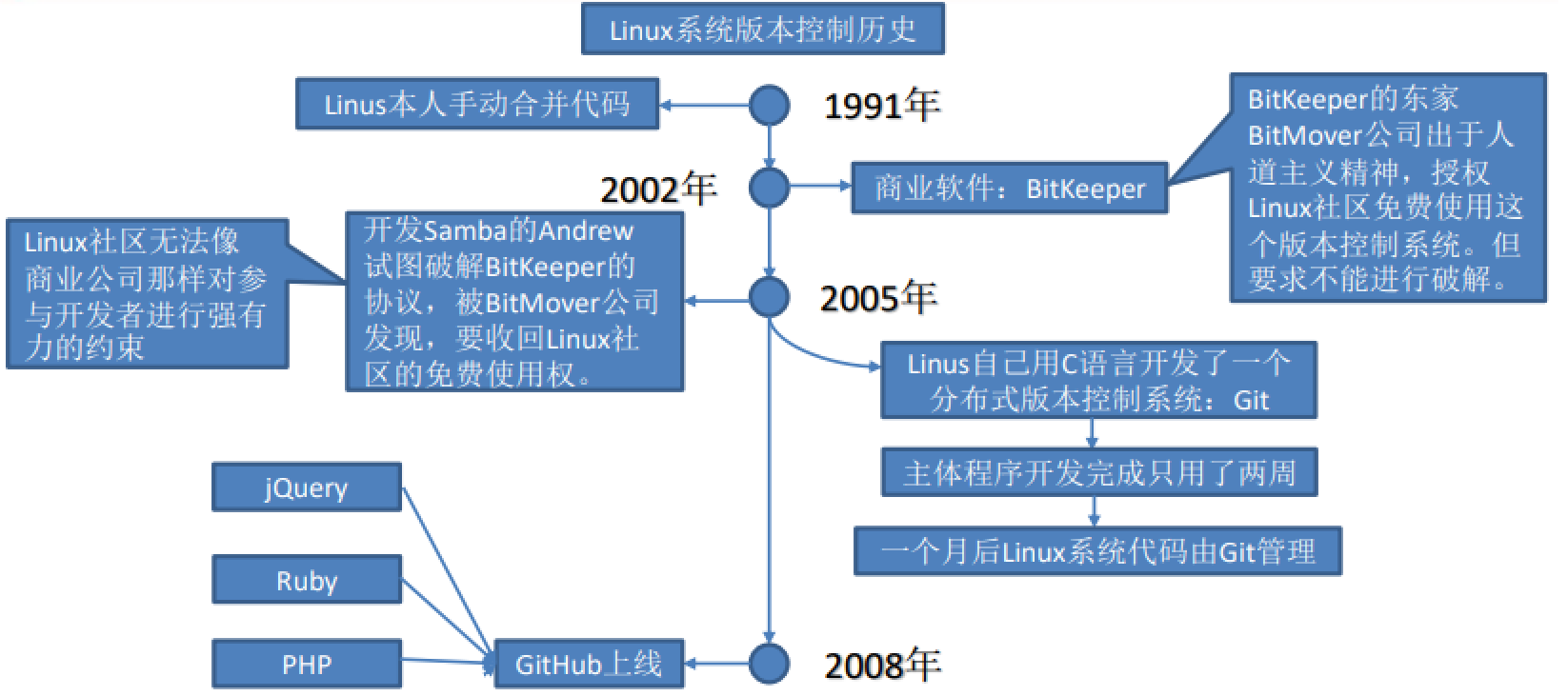
…

集成idea

## GitLab 局域网的代码托管中心

# Git概述

1. Git 是一个免费的、开源的分布式版本控制系统，可用作于小型到大型的各种项目。
2. Git 易于学习，性能强，效率高。 C语言写 linux
3. Git具有廉价的本地库，方便的暂存区域和多个工作流分支等特性。
4. Git性能优于 CVS、Subversion、Perforce等其他版本控制工具。



集中式的版本控制系统 SVN

分布式的版本控制系统 Git





# Git的工作机制

1. 工作区 （写代码的地方，认为是idea中创建的一个项目）

add 添加

1. 暂存区 （git可以追踪到暂存区的文件，这里的代码可以修改）

commit 提交本地库 形成一个新的版本，这个新的版本是不能再做修改的！如果要

修改只能在该版本基础之上修改，修改之后又形成了一个新的版本。

v1.0

修改

v2.0

1. 本地库 不能修改

push

1. 远程仓库 GitHub Gitee Gitlab

# Git的下载和安装

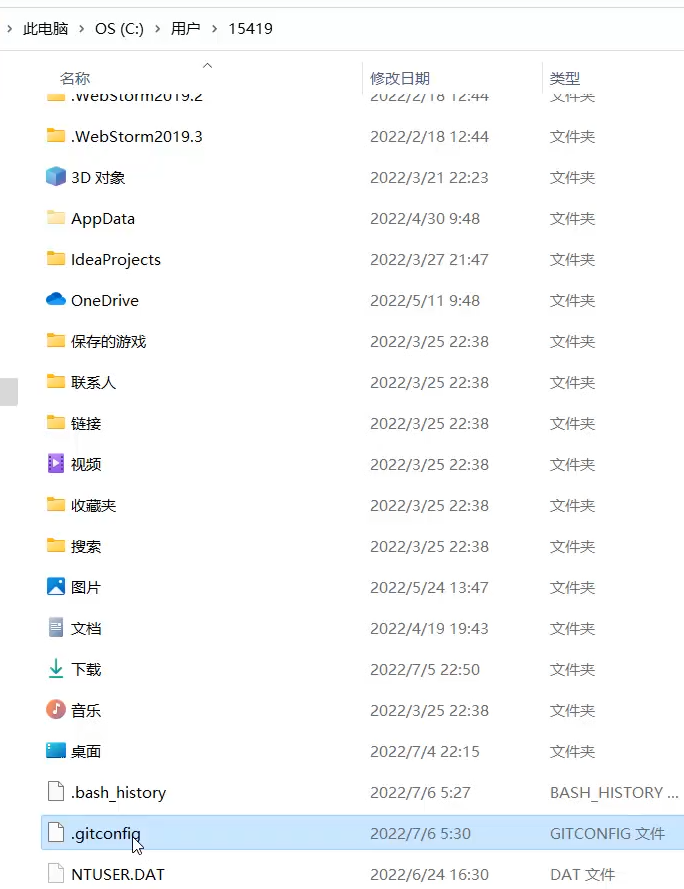
官网

[**git**-scm.com](http://www.baidu.com/link?url=rSd91oJYlFYmlXO6njQlW3e04JJjwI-DSZOMq9lD1zi)

# Git的常用命令

|  |  |
| --- | --- |
| 命令名称 | 解释 |
| **git config --global user.name 用户名称** | **设置用户名 （必须要设置，否则无法提交代码）** |
| **git config --global user.email 用户邮箱** | **设置用户邮箱（必须要设置，否则无法提交代码）** |
| Git init | 初始化本地库 |
| Git status | 查看本地库状态 |
| Git add 文件名称 | 添加到暂存区 |
| Git commit -m “日志信息” 文件名 | 提交到本地库 |
| Git reflog | 查看历史记录 |
| Git reset –hard 版本号 | 版本穿梭 |





# 初始化本地库

git init

# 查看本地库状态

git status

# 添加至暂存区

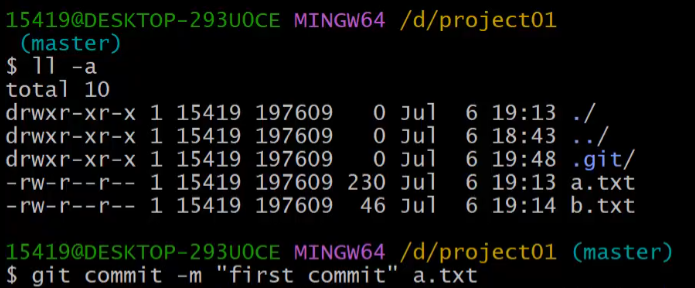
git add 文件名

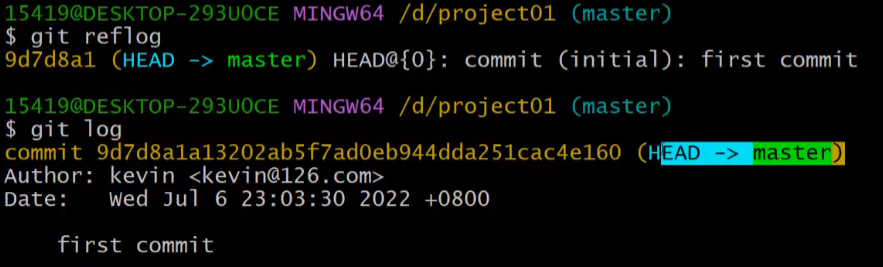
当删除暂存区的文件之后，该文件在工作区依旧存在（没有被git追踪到）！

git rm –cached 文件名称

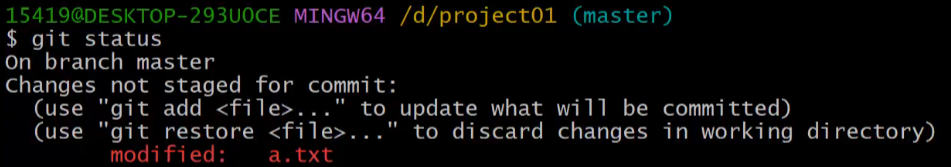
# 提交本地库

git commit -m "备注" 文件名称

6



# 修改文件





# 版本穿梭

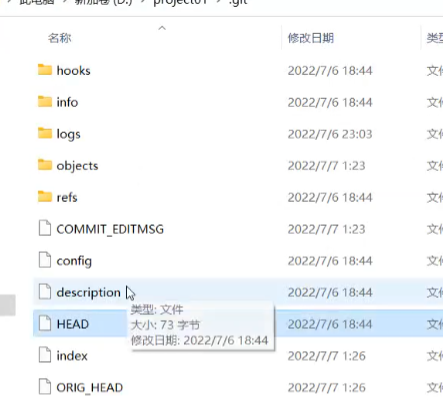
sasafs1.0 a.txt

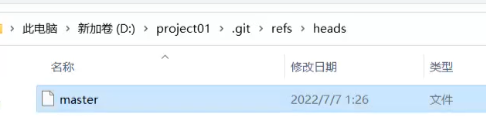
sfsfs2.0 a.txt

sfssfss3.0 a.txt

git reset --hard 版本号





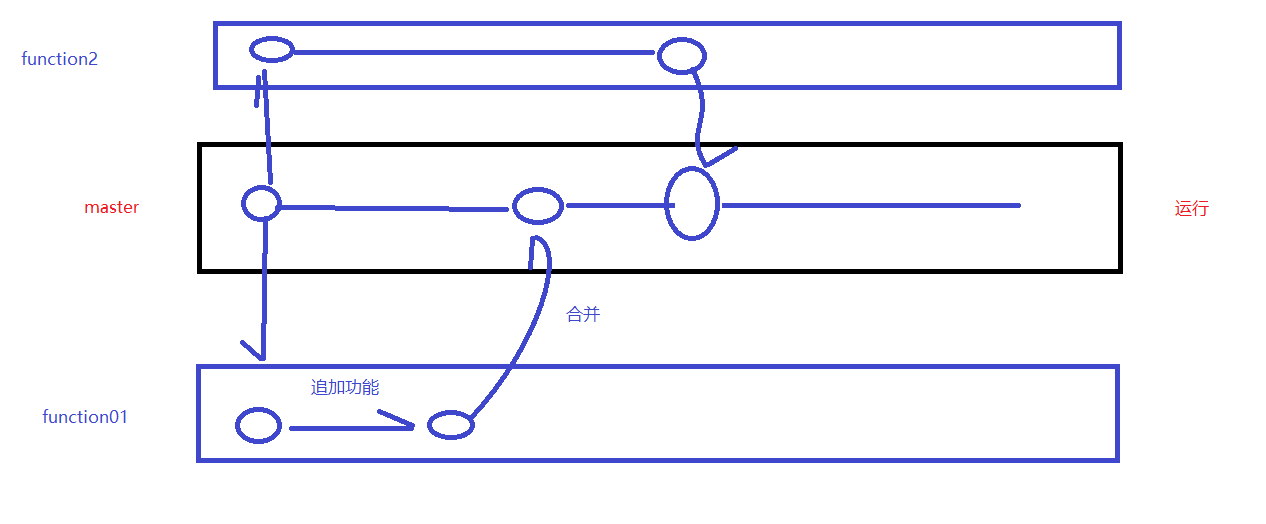


# 什么是分支

简单的理解，分支其实就是副本的意思!

master

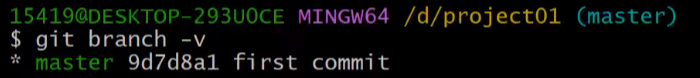
在版本控制开发的项目过程中，可能会有多个任务同时推进，为每个任务，我们就可以创建一个单独的分支。使用分支就意味着我们可以把自己的工作从开发主线上分离出来，开发自己分支的时候，不会影响主线分支的运行。开发完成之后，可以合并分支，完成项目功能的添加。



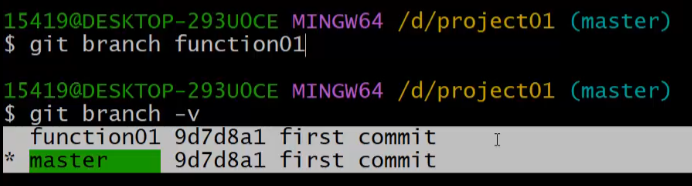
# 操作分支的常见命令

|  |  |
| --- | --- |
| 命令名称 | 作用 |
| git branch 分支名 | 创建分支 |
| git branch -v | 查看分支 |
| git checkout 分支名 | 切换分支 |
| git merge 分支名 | 把指定的分支合并到当前分支上 |

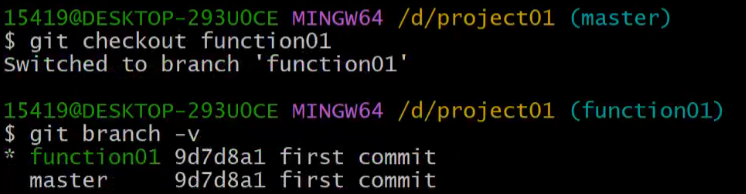
**查看分支：**



**创建分支：**



**切换分支：**

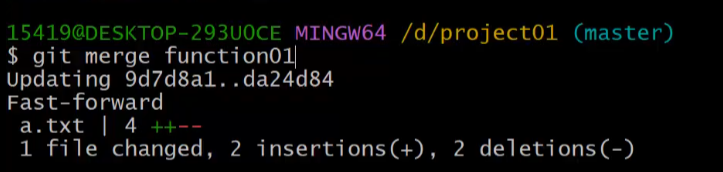


# 合并分支

git merge 分支名

master \*\*\*

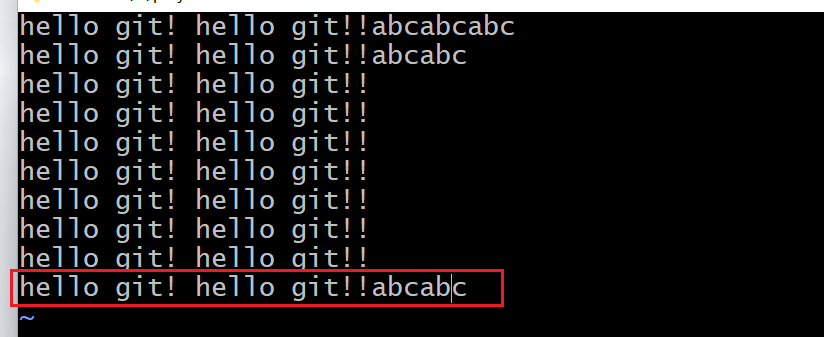
function01



# 合并分支中遇见冲突的解决

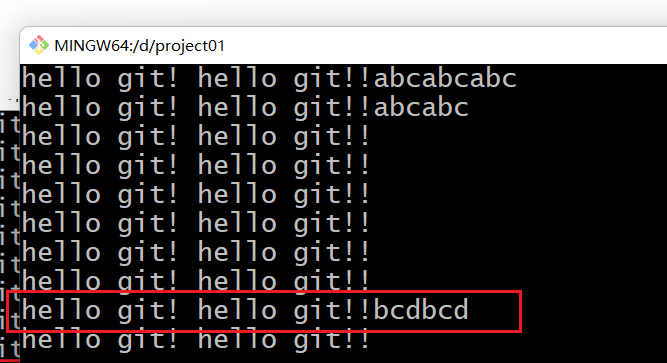
function01

1. txt



master

1. txt



问：

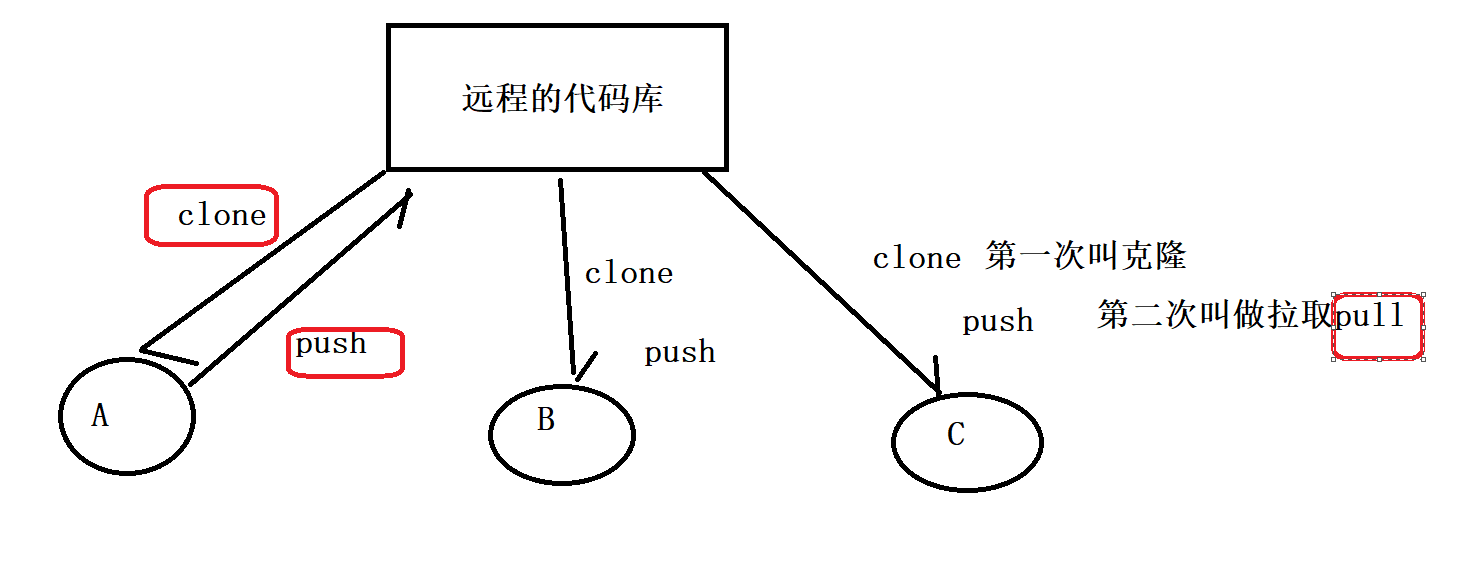
在master分支下合并function01 a.txt

# Git的团队协作机制

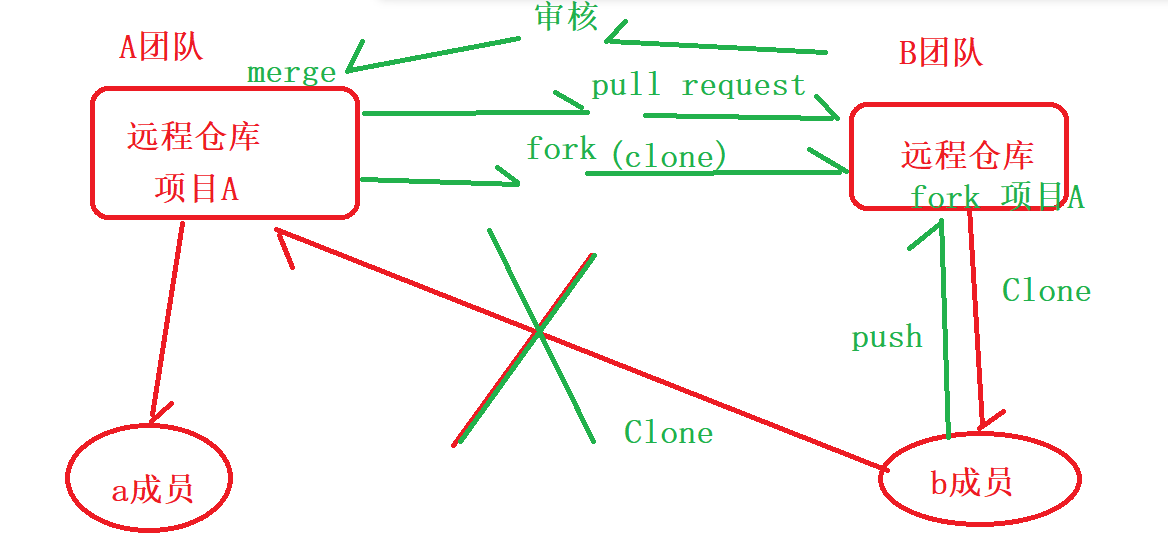
团队内协作 共享一个远程库

跨团队的协作 有多个远程库

团队内协作



跨团队协作



# 创建远程仓库

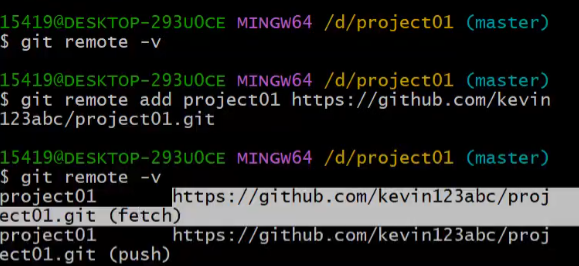
github.com

<https://github.com/kevin123abc/project01.git>

# 创建别名和查看别名

git remote -v 查看别名

git remote add 别名 远程仓库地址



# 推送本地库到远程库

git push 别名/url 分支名称



<https://blog.csdn.net/chuyouyinghe/article/details/122203309>

ghp\_jLhnpK079K4vbAzHJX21h8uKyHJSa62sCzXo

git remote set-url project01 https://ghp\_jLhnpK079K4vbAzHJX21h8uKyHJSa62sCzXo@github.com/kevin123abc/project01.git

<your\_token>：换成你自己得到的token

<USERNAME>：是你自己github的用户名

<REPO>：是你的仓库名称

# 拉取远程库到本地库

git pull 远程仓库名称 分支名称



# 克隆远程仓库

git clone 远程仓库地址

public

从远程仓库克隆的仓库

1. 已经帮咱们把本地仓库初始化了
2. 帮咱们给远程仓库去取了别名 origin

# 团队内成员推送本地库

# 跨团队协作

[154199755@qq.com](mailto:154199755@qq.com) s4632727t A团队

[2189338722@qq.com](mailto:2189338722@qq.com) kevin123abc B团队

团队合作：

第一步：A团队 fork b团队

# SSH免密登录

ssh-keygen -t rsa -C 登录账户名称

# IDEA集成 GIT的环境配置

## 忽略项目中不相干文件

1. 创建忽略规则文件 某某.ignore（名字可以随便起，建议是 git.ignore，但后缀需要是ignore）
2. 2)文件的存放位置建议放在当前系统用户的家目录，因为这样可以便于让~/.gitconfig 文件引用。

git.ignore 文件模版内容如下：

|  |
| --- |
| # Compiled class file  \*.class  # Log file  \*.log  # BlueJ files  \*.ctxt  # Mobile Tools for Java (J2ME)  .mtj.tmp/  # Package Files #  \*.jar  \*.war  \*.nar  \*.ear  \*.zip  \*.tar.gz  \*.rar  # virtual machine crash logs, see  http://www.java.com/en/download/help/error\_hotspot.xml  hs\_err\_pid\*  .classpath  .project  .settings  target  .idea  \*.iml |

1. 在.gitconfig 文件中引用忽略配置文件

[user]

name = Layne

email = [kevin@126.com](mailto:kevin@126.com)

[core]

excludesfile = C:/Users/1541/git.ignore

注意：这里要使用“正斜线（/）”，不要使用“反斜线（\）”

## idea集成git

# idea中git的相关基础操作

初始化本地库 git init

添加暂存区

提交本地库

# 切换版本

# 分支的创建和切换

# 正常合并分支

# 冲突合并分支

# idea中登录github账号

# idea中将本地库推送(分享)到远程库

# 将本地库更新推送到远程库

# 从远程库拉取代码

注意：一般情况下，将本地库推送到远程库的时候要求本地库版本要高于远程库！所以咱们在push之前都会先pull远程代码库！

# 从远程代码库克隆代码到本地

# 码云介绍

gitee

# idea集成码云

# 将本地库分享到码云

拉取的前提：

1. 从码云把仓库克隆到本地

或

1. 本地的代码分享到了码云

# 从github导入库到码云