版本控制工具的使用，多人协作工具 Version Control System （VCS）

企业开发：源码安全、多人并行开发

使用工具：

SVN：集中式版本控制系统（需要有一个服务器存储完整代码）

一旦服务器崩溃整个版本控制系统瘫痪

GIT：分布式版本控制系统（没有明确的服务器概念，每个客户端都可以充当服务器）

仓库：存放项目源码，素材，文档

克隆：第一次从服务器下载项目

提交：commit 将本地更新（添加，删除，修改）同步到服务器

更新：将服务器更改同步到本地

冲突：自己更新，服务器也更新了同一个文件，产生冲突

GIT命令使用

GIT操作位置：再磁盘根目录新建空文件夹 Py1904

1. 初始化仓库（需要在空的文件夹）

Git init 可以初始化一个仓库

1. 查看状态

Git status

1. 创建用户身份

全局：系统盘、用户、当前用户、.gitconfig配置项

Git config –-global user.email \*\*\*

Git config –-global user.name \*\*\*\*

项目用户：配置文件位于当前项目目录 .git / config

Git config user.email \*\*\*

Git config user.name \*\*\*\*

1. 工作区到暂存区

Git add文件名 文件名 文件名 文件夹

1. 暂存区到工作区

Git reset HEAD 文件名

1. 撤销工作区修改

Git checkout – 文件名

1. 提交本地仓库

Git commit -m 备注 （将暂存区中所有修改一次性同步）

1. 查看日志

Git log

Git reflog

1. 回档、回到历史版本

Git reset --hard 版本号

Git reset --hard HEAD^

Git reset --hard HEAD~n

1. 分支：多个模块并行开发

一个分支针对于单独模块

Git branch 查看所有分支

Git branch name新建分支

Git checkout name 切换分支

Git merge name 分支合并

Git branch -d name 删除分支

Git checkout -b name 创建并且切换分支

1. 参与项目组开发

由项目经理创建仓库并且分发仓库地址

项目组成员通过仓库地址获取项目

Git clone 地址

当两个分支操作到同一个文件中的同一行，

如果合并两个分支，

则会自动合并失败

需要手动解决冲突（删除==== <<<<<< >>>>>>）

然后提交

在没有.git管理的目录使用git命令

报错 not a git repository

使用git命令需要处于.git同级目录或者子目录

在Pycharm中使用GIT插件

1. 使用open打开项目所在目录，确保.idea和.git同级目录
2. 加入忽略列表：在.git同级目录有 .gitignore

将需要忽略的文件在其中编写忽略规则

1. 回档流程：
   1. 进入 version control 切换到log标签
   2. 选中某一个版本，右键拷贝版本号
   3. 选中项目根目录 右键-》GIT-》Respository-》Reset Head -》选中hard
   4. 需要注意 只能管理推送到远程仓库的版本

