SVN:Subversion，集中式版本控制

终端输入：

1.上传文件到svn服务器： svn+import+文件本地路径+服务器路径 +（–m)+ “提交的备注信息（方便别人查看）”

2.服务器下载文件到本地： svn+checkout+文件服务器路径+本地保存路径 --username=xxx --password=xxx

3.提交修改的文件：进入文件的路径，svn+commit+（–m)+ “提交的备注信息”

4.新添加文件：进入文件的路径，

svn+add+文件1+文件2+… ,最后要commit

5.删除文件(一定不要手动删除)：进入文件的路径，

svn+remove+文件1+文件2+… ,最后要commit

6.更新 svn+update

7.解决冲突：svn+resolve+冲突的文件名

系统有三种方案：

p:延迟处理，会生成3个文件，冲突代码合并，需要执行 svn resolve filename

mc:使用我的代码

tc:使用他人的代码

8.版本回退：

修改但还没提交：svn+revert+要回退的文件名（返回上一个版本的状态）

修改并已经提交：首先 update到最新版本

然后 svn+merge+文件名+(-r)+版本1：版本2

最后 svn+commit(从版本1会退到版本2)

9.查看本地路径是否修改：进入文件的路径，svn+status

10.查看svn日志 svn+log

11.加锁 svn+lock

12.解锁 svn+unlock

<快捷 ci---commit ; st---status; up---update>

svn目录规范：

trunk:主干，当前开发项目的主目录

branch:分支目录，添加非主线的功能，修改bug，开发新技术

tag:标记目录，通常作为重大版本的备份

在trunk开发完成1.0版本，然后在tag添加1.0版本的备份，接着继续开发2.0版本，这时候测试发现1.0版本有bug，然后在tag开启一个branch分支，当bug修复完成变成1.1版本，同时在tag备份1.1版本，而且把branch的1.1版本合并到trunk中

svn第三方托管平台：SVNSpot

！：文件丢失（手动删除）

GIT:分布式源代码管理

1.多人开发需要一个共享版本库，单人开发初始化一个本地库即可

2.本地共享库包括：

本地共享库：文件夹、U盘、硬盘

远程共享库：搭建的git服务器、第三方（gitHub、oschina）

Git和Svn对比：

Git 速度 > SVN

Git是分布式管理，SVN是集中式管理

Git可以轻松拥有无限个分支，SVN使用分支比较笨拙

Git支持本地版本控制工作，SVN必须联网才能正常工作

Windows在控制台输入ipconfig获取ip地址

Com+space 快速查找系统软件或文件

输入 “终端” ---- 快速打开终端

com+k --- 清空终端信息

终端快捷键：

touch xxx.xxx --- 新建文件

rm xxx.xxx --- 删除文件

open xxx.xxx --- 打开文件

mkdir xxx --- 新建文件夹

mv xxx.xxx 路径 --- 移动文件

pwd --- 显示文件所在路径

新建 点开头的为隐藏文件，如 .doc

HTTP状态码的含义：

2xx （成功）

200（成功）：服务器已成功处理了请求

202（已接受）：服务器已接受请求，但尚未处理

203（非授权信息）

204（无内容）：服务器成功处理了请求，但没有返回任何内容。

3xx （重定向)

301（永久移动）：请求的网页已永久移动到新位置

302（临时移动）

4XX（请求错误）

400（错误请求）：服务器不理解请求的语法

401（未授权）：请求要求身份验证

403（禁止）：服务器拒绝请求

404（未找到）：服务器找不到请求的网页

408（请求超时）

410（已删除）：请求的资源已永久删除

413（请求实体过大）

414（请求的 URI 过长）

5xx（服务器错误）

500（服务器内部错误）

502（错误网关）

503（服务不可用）：服务器目前无法使用（由于超载或停机维护）。通常，这只是暂时状态。

504（网关超时）

505（HTTP 版本不受支持）