GIT课程

1. 什么是git？

a.Git就是源代码管理工具

b.什么是源代码？在一个项目中，凡是由开发人员编写的代码，都是源代码

c.为什么要管理源代码？为了让源代码可以被追溯，通过git，可以知道哪个开发人员变更了代码，变更了哪些代码，一旦现有版本出现重大bug，可以回退到原来的版本

d.GIT是Linux之父当年为了维护管理Linux的源代码而编写的一个工具，git之前有很多管理工具svn/vss/tfs/hs…

e.git相关技术文档

<https://git-for-windows.github.io下载git>

git-scm.com git官方网站

http://zccst.iteye.com/blog/2191589 git使用原理http://www.ruanyifeng.com/blog/2015/12/git-cheat-sheet.html阮一峰的网络日志

http://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/0013743256916071d599b3aed534aaab22a0db6c4e07fd0000 廖雪峰的官网

2. GIT安装

1)git命令行工具安装：

根据计算机系统来选择安装的x32位、x64位的git，默认安装即可

2)配置环境变量：

控制面板>系统和安全>系统>高级系统设置>环境变量>新建》

I 变量名：GIT\_HOME

变量值：D:\Program Files (x86)\Git(看到bin包为止)

即git安装目录:D:\Program Files (x86)\Git

Ii 配置path：%GIT\_HOME%\bin;(如果前面没有;加;)

然后点击确定》确定

3)检测git是否安装成功

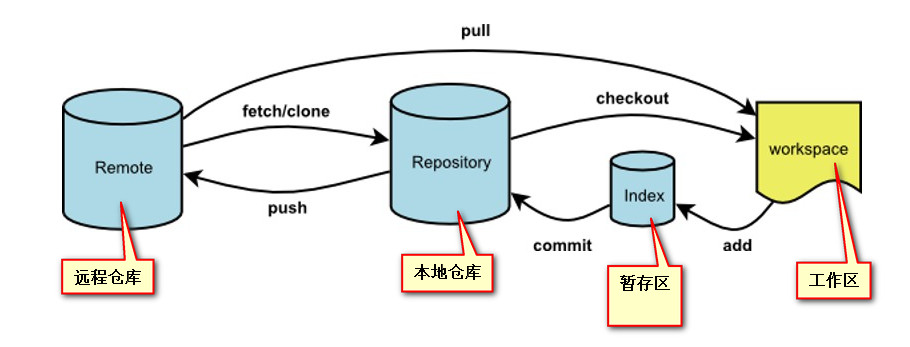
Win+R调出命令行：cmd

git - -version 回车键 打印git版本号

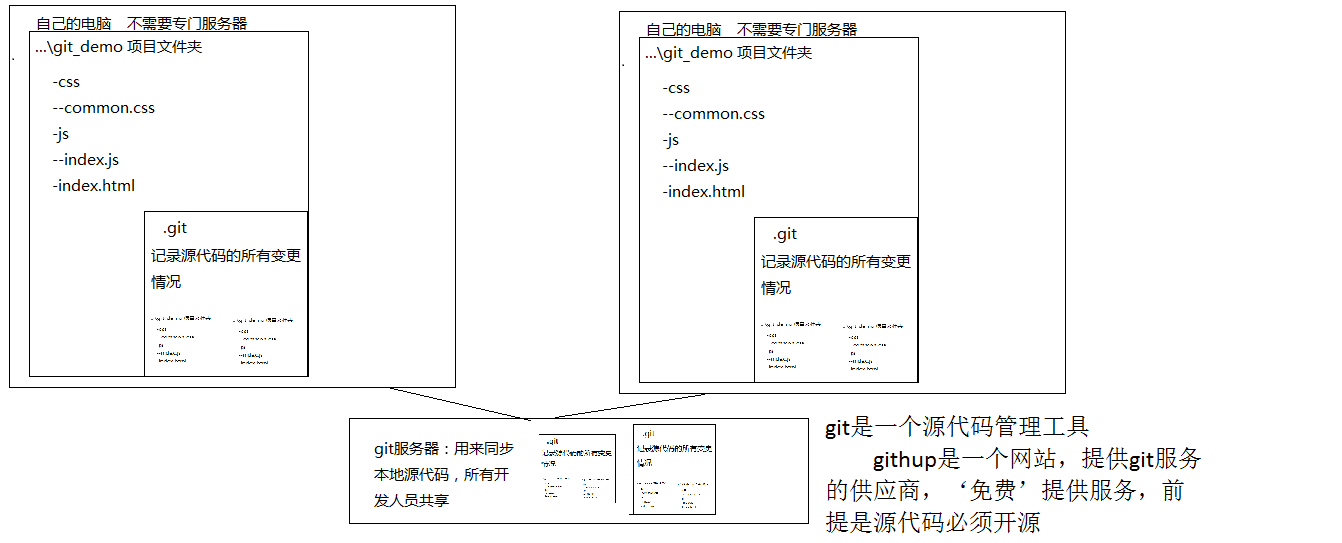
另外，基于git命令行的一个客户端可视化操作软件sourcetree <https://www.sourcetreeapp.com/>

安装：

对于git管理代码理解



staged



在根目录www下创建git\_demo文件夹，创建一个web项目，然后shift+鼠标右键》在此处打开命令窗口

3. GIT命令操作

pwd ：查看当前所在目录；

cd:切换目录(change directory)，后接要切换到的目录，

例如从c盘切换到d盘；cd /d

打开一个有多个字符命名的文件cd ‘progress a’

cd .. 返回上一级目录 cd ../ 和 cd ./ 打开当前目录

dir：显示当前目录下所有文件

ls（list命令）：显示当前目录下所有的文件；

ls -a（list-all命令）：显示当前目录下所有的文件，包括隐藏的目录；

mkdir:创建文件夹 mkdir .net

vim:创建文件 vim demo.html vim css/css.css

:wq write quit(退出进入的文件)

:q 直接放弃

a/i 编辑文件

Esc键 退出编辑

# 注释

cd:进入文件夹 cd hello

echo ‘’>.gitignore 创建文件

rm：删除文件（remove）别名：del

rm -r：删除目录（remove recursively递归）

clear 清空当前控制台 别名：cls

ctrl+c 终止当前执行的命令

git help 打印所有的git命令 ，并简短解释

git help - a 打印所有的git命令

git help add

git　help add >> 1.txt 将add简介写入到1.txt文件

a.初始化一个本地GIT仓储

git init 初始化一个本地的仓库，就是在本地文件夹中添加了一个.git文件夹，用于记录所有的项目变更信息

Initialized(初始化) empty Git repository(仓库) in D:/phpStudy/WWW/git\_demo/.git/

注意：如果没有.git文件，通过查看》选项》查看》取消勾选"显示隐藏的文件.文件夹和驱动器",点击确定就可查看隐藏文件

b.查看本地仓储的变更状态

git status ：用于查看本地仓储的状态，第一次查看，显示的是所有的文件没有被跟踪

Untracked files:未被跟踪的文件

git status -s 输出简要的变更日志 s:simple

c.添加本地暂存(托管)文件

git add 可以将未被跟踪的文件添加到跟踪列表

git add 文件名/文件夹名

git add . 追踪所有被忽略的文件

d.添加本地GIT忽略清单文件

官网参考：github.com/github/gitignore

在代码库文件夹的根目录添加一个.gitignore文件，

a/i 进入编辑

Esc键 退出编辑

直接书写要忽略的文件：

#忽略template\_c中自动生成的编译文件

\*.tpl.cache.php

#忽略cache中自动生成的缓存文件

\*.tpl.php

#忽略runtime中的log日志

webroot/protected/runtime/application.log

此文件用于说明忽略的文件有哪些

如何创建？

window不支持以点开头的文件和文件夹名称：

只有使用命令行-mkdir .aa 创建.aa文件夹 echo ‘’>.gitignore 创建.gitignore文件

git add . 或者git add --all

e.提交被托管的文件变化到本地仓储

git commit –m ’第一次提交源代码’ //m:message

将工作区文件的变化提交到本地仓库，一般工作中实现一个功能单元，再commit

f.设置用户名和邮箱

git config - -global user.name "username"

git config - -global user.email "email"

git config --list

git config - -unset - -global user.name

cat ~/.gitconfig ~表示当前用户的主目录

g.对比版本差异

git diff 查看工作区代码与暂存区的不同

git diff --staged 查看本地仓库和暂存区不同

h.查看提交日志

git log

git log --oneline 以oneline形式显示当前分支

git log --oneline --all 不管分支，都显示

-5 显示五条记录

--graph 图形展示

--grep=”index.html”查看指定文件log

--before=”2014-08-09”

“1 week” “4 days”

1. renamed 文件

方法一：手动重命名文件

git rm test.html:注意只有被commit完成以后并且没有任何修改的文件才能被删除

git add demo.html

git status

git commit -m “”

方法二：直接用git mv :只要文件被add以后但是不能commit就可以执行mv

文件重命名

git mv test.html demo.html

git commit –m “”

j.文件移动

git mv test.html test

git commit –m “”

文件夹移动

git mv test html/

git commit –m “”

k.删除、修改的文件恢复

k1删除命令未commit，恢复文件

git rm test.txt

git checkout HEAD -- test.txt 将test.txt文件恢复到最近的一次提交状态

其中HEAD指向最近的一次commit，--表示当前分支

HEAD^最近一次提交的上一次提交

HEAD^^最近一次提交的上两次提交

HEAD~10最近一次提交的上十次次提交

K2. 删除文件并commit该操作，

git rm test.txt

git commit -m "delete test.txt"

git checkout HEAD^ - - test.txt

用 git rm 来删除文件，同时还会将这个删除操作记录下来；

用 rm 来删除文件，仅仅是删除了物理文件，没有将其从 git 的记录中剔除。

直观的来讲，git rm 删除过的文件，执行 git commit -m "abc" 提交时，会自动将删除该文件的操作提交上去。

而用 rm 命令直接删除的文件，单纯执行 git commit -m "abc" 提交时，则不会将删除该文件的操作提交上去，需要在执行commit的时候，多加一个-a参数，

即rm删除后，需要使用git commit -am "abc"提交才会将删除文件的操作提交上去。

比如：

1）删除文件test.file

# git rm test.file

# git commit -m "delete test.file"

# git push

或者

# rm test.file

# git commit -am "delete test.file"

# git push

2）删除目录work

# git rm work -r -f

# git commit -m "delete work"

# git push

l.revert:恢复某一次提交之前的状态

git log - -oneline

git revert 068dfc9

m.reset

一般commit以后HEAD指针默认指向最后一次提交，但通过reset可以让HEAD指针指向之前某一次提交，这样就会覆盖之后的所有提交

reset - -soft 软重置 不会影响重置前工作区和暂存区的代码的状态 git reset --soft 6caf478

reset - -mixed reset默认状态 ，会将暂存区重置到工作区的状态，并把指针指向该次提交

reset --hard 会将工作区和暂存区重置到reset指定的状态

n. git diff master..zh\_branch 对比两个分支的不同

git diff master..zh\_branch index.html对比两个分支某一文件的不同

o.git merge zh\_branch将zh\_branch分支合并到master

1)Fast-forward合并(只有zh分支改变)：直接将zh\_branch合并到master,不会产生新的提交

2)主次分支都改变了不同的地方：

git commit –am “” add和commit合并

git log --oneline - -all -5 --graph

git merge zh\_branch

3)主次分支都改变了相同的地方：

git merge zh\_branch

出现冲突：1）git merge --abort 放弃解决冲突

2)解决冲突

<<<<<<< HEAD

<title>我是在mastera分支修改</title>

=======

<title>我是在zh\_branch分支修改</title>

>>>>>>> zh\_branch

i红色去掉，绿色二选一

ii从新提交 git add . git commit –m “”

p. git branch –m zh\_branch zhang\_branch 重命名分支

git branch –d zhang\_branch 删除分支

r. git clone <https://github.com/zh/demo.git> demo\_zh

s． git fetch:将远程库origin master分支的代码拉到本地库的origin/master分支，然后git merge origin/master

4.github与git区别

git是一个源代码管理工具

github是一个网站，提供git服务的供应商，‘免费’提供服务，前提是源代码必须开源

5.GITHUB基本使用

<Https://github.com>

社交化编程

<https://github.com/jobbole/awesome-javascript-cn>

<https://github.com/jobbole/awesome-css-cn>

<https://github.com/alibaba>

<https://github.com/fex-team>百度前端研发部

1. 注册账号
2. $ git remote add origin https://github.com/Ocean490724726/git\_demo.git
3. git remote不带参数，列出已经存在的远程分支
4. git remote -v (verbose) 列出详细信息，在每一个名字后面列出其远程url
5. git pull origin master
6. git remote rm origin 删除远程所有连接

6.GIT分支

a. git branch 获取本地仓库分支

git branch -r 查看远程分支

git branch -a 查看所有的分支(含远程分支)

b.git branch zh\_branch 添加本地仓库分支

c.git checkout zh\_branch 切换本地仓库分支

d.git push -u origin zh\_branch 将zh\_branch分支推送到远程origin分支,并跟踪远程分支origin/zh\_branch的变化，

如果不加u，git就不会追踪远程分支的变化

e.git branch –m zh\_branch zhang\_branch 重命名

f.git branch -d zhang\_branch 删除

push到远程仓库

7.通过创建gh-pages分支可以在网页中查看

<https://pages.github.com/>创建页查看说明

1. 创建gh-pages分支,并提交到githup上
2. 在git上查看settings

本页面下面有删除仓库功能

1. 绑定域名：在你购买域名的网站下进行域名解析
2. 在根目录下，创建CNAME文件,在该文件里面书写你要绑定的域名，将该文件提交到远程仓库

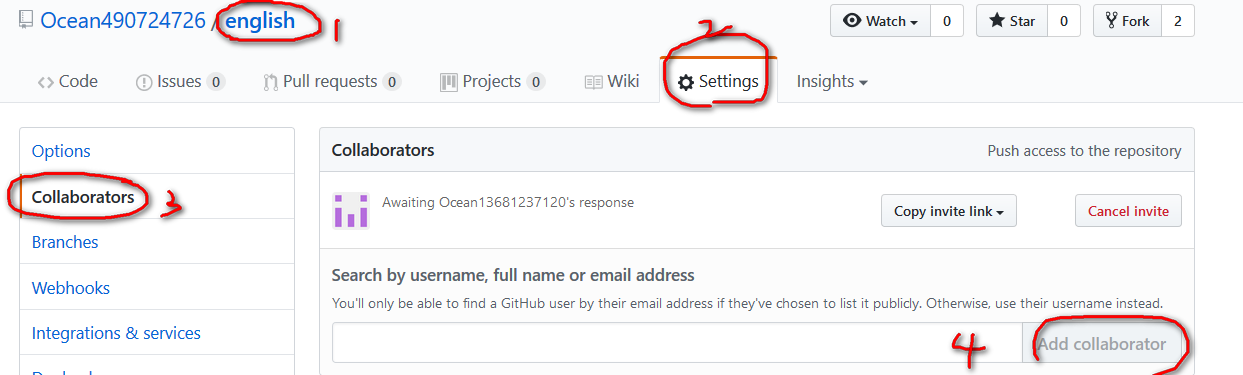
Download:下载源代码

Git clone 下载版本库，可以看到每次commit，并将自己开发的代码提交到本地仓库

fork：基于该版本，开发自己的项目，并提交的远程仓库，先fork，然后clone到本地，然后继续开发，最后push到自己的远程仓库

pull request:先fork，如果b想将自己开发的代码分享给最初开发者a，可以申请pull request,经过a审核同意，即可将b的代码合并到a的源代码仓库

contributors:添加项目贡献者b，b就有权限直接将自己开发的代码push到项目发起人a的远程代码库

具体：项目发起人a在如下图添加共同开发b人员

然后b收到一份a添加b为项目开发人的邮件

此时，b就可以clone 该项目，然后开发并push到a的远程仓库

报错1：

To github.com:Ocean490724726/student\_enlish.git

! [rejected] gh-pages -> gh-pages (fetch first)

error: failed to push some refs to 'git@github.com:Ocean490724726/student\_enlish.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

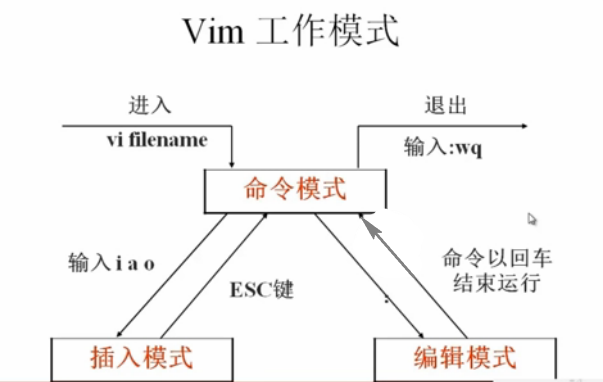
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

解决方法：$ git push -f origin gh-pages

其中-f：force强制提交

Vim文本编辑器

Vim是linux/unix最常用的文本编辑器，它的作用是建立、编辑、显示、删除文本文件，vim没有菜单，只有命令



插入命令

|  |  |
| --- | --- |
| a | 在光标所在字符后插入 |
| A | 在光标所在行尾插入 |
| i | 在光标所在字符串前插入 |
| I | 在光标所在行首插入 |
| o | 在光标下插入新行 |
| O | 在光标上插入新行 |

注意：在使用插入命令之前，要先Esc，进入命令模式

定位命令

|  |  |  |  |
| --- | --- | --- | --- |
| 命令 | 作用 | 命令 | 作用 |
| :set nu | 设置行号 | :n | 到第n行 |
| :set nonu | 取消行号 | $ | 移至行尾 |
| gg  G | 到第一行  到最后一行 | 0（零） | 移至行首 |
| nG | 到第n行 | 5G |  |

删除命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| X | 删除光标所在处字符 |
| nx | 删除光标所在处后n个字符 |
| dd | 删除光标所在行，ndd删除n行 |
| dG | 删除光标所在行到文件末尾内容 |
| D | 删除光标所在处到行尾内容 |
| :n1,n2d | 删除指定范围的行 |

复制和剪切命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| yy | 复制当前行 |
| nyy | 复制当前行以下n行 |
| dd | 剪切当前行 |
| ndd | 剪切当前行以下n行 |
| p、P | 粘贴在当前光标所在行下或者行上 |

就是yy和p或者dd和p的组合键

替换和取消命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| r | 取代光标所在处字符 |
| R | 从光标所在处开始替换字符，按Esc结束 |
| u | 取消上一步操作 |

搜索和搜索替换命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| /string | 搜索指定字符串,按Enter键  搜索时忽略大小写:set ic  区分大小写:set noic |
| n | 搜索指定字符串的下一个出现位置 |
| :%s/old/new/g | s是substitute替换，g是global， %是第一行的意思，c是confirm  g:替换时询问  c:替换时询问确认  全文替换指定字符串 |
| :n1,n2s/old/new/g | n1:起始行 n2:终止行  在一定范围内替换指定字符串 |

保存和退出命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| :w | 保存修改 |
| :w new\_filename | 另存为指定文件 |
| :wq | 保存修改并退出 |
| ZZ | 快捷键，保存修改并退出 |
| :q! | 不保存修改退出 |
| :wq! | 保存修改并退出(文件所有者及root可使用) |

导入命令执行结果 :r

!命令

定义快捷键：map 快捷键触发命令

范例： :map ^P I#<ESC>

:map ^B 0x

连续行注释：:n1,n2s/^/#/g

:n1,n2s/^#//g

:n1,n2s/^/\/\//g

替换： :ab mymail 490724276@qq.com

Linux课程

英文提示的困惑

Command not found

No Such file or directory

忘掉Windows思维方式：让中年家庭妇女都会使用

Windows：人性化界面

Linux :稳定、安全

[.常见指令](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t1)  
[2.系统管理命令](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t2)  
[3.打包压缩相关命令](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t3)  
[4.关机/重启机器](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t4)  
[5.Linux管道](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t5)  
[6.Linux软件包管理](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t6)  
[7.vim使用](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t7)  
[8.用户及用户组管理](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t8)  
[9.文件权限管理](http://myblog.wsz7777.cn/2017/08/15/linux-%e5%b8%b8%e7%94%a8%e5%91%bd%e4%bb%a4/#t9)

**常用指令**

***ls*** 显示文件或目录

***-l*** 列出文件详细信息

***-a*** 列出所有文件目录

***mkdir*** 创建目录

***-p*** 创建父级目录，若无父级目录则正常创建目录

***cd*** 切换目录

***touch*** 创建空文件

***echo*** 创建带有内容的文件

***cat***  查看文件内容

***cp*** 拷贝

***mv*** 移动或重命名

***rm*** 删除文件

***-r*** 递归删除，可删除子目录及文件

***-f*** 强制删除

***find*** 搜索文件

***wc*** 统计文本中行数，字数，字符数

***grep*** 在文本中查找某个字符串

***rmdir*** 删除空目录

***tree*** 树形结构显示目录，需要安装tree包

***pwd*** 显示当前目录

***ln*** 创建连接文件

***more/less***   分页显示文本文件内容

***head/tail*** 显示文件头、尾内容

***ctrl+alt+F1*** 命令行全屏模式

**系统管理命令**

***stat*** 显示指定文件的详细信息

***who*** 显示在线登录用户

***whoami*** 显示当前操作用户

***hostname*** 显示主机名

***uname*** 显示系统信息

***top*** 动态显示当前耗费资源最多的进程信息

***ps*** 显示瞬间进程状态（ps -aux）

***du*** 查看目录大小

***-h/home*** 带有单位显示目录信息

***df*** 查看磁盘大小

***-h*** 带有单位显示磁盘信息

***ifconfig*** 查看网络状况

***ping*** 测试网络连通

***netstat*** 显示网络状态信息

***man*** 命令不会用就找他（eg：man ls）

***clear*** 清屏

***alias***         对命令重命名 如：alias showmeit="ps -aux" ，另外解除使用unaliax showmeit

***kill***          结束进程（可以先用ps 或 top命令查看进程的id，然后用kill命令结束）

**打包压缩相关命令**

***gzip:***

***bzip2:***

***tar:*** 打包压缩

***-c*** 归档文件

***-x*** 压缩文件

***-z*** gzip压缩文件

***-j*** bzip2压缩文件

***-v*** 显示压缩或解压缩过程 v(view)

***-f*** 使用档名

例:

***tar -cvf /home/wsz.tar /home/wsz*** 只打包，不压缩

***tar -zcvf /home/wsz.tar.gz /home/wsz*** 打包，并用gzip压缩

***tar -jcvf /home/wsz.tar.bz2 /home/wsz*** 打包，并用bzip2压缩

（注：解压时把 c 换成 v 即可）

**关机/重启机器**

***shutdown***

***-r*** 关机重启

***-h*** 关机不重启

***now*** 立刻关机

***halt*** 关机

***reboot*** 重启

**Linux管道**

将一个命令的标准输出作为另一个命令的标准输入。  
也就是把几个命令组合起来使用，后一个命令除以前一个命令的结果。

***grep -r “close” /home/\* | more***

在home目录下所有文件中查找，包括close的文件，并分页输出。

**Linux软件包管理**

dpkg (Debian Package)管理工具，软件包名以.deb后缀。这种方法适合系统不能联网的情况下。比如安装tree命令的安装包，先将tree.deb传到Linux系统中。再使用如下命令安装。

***sudo dpkg -i tree\_1.5.3-1\_i386.deb*** 安装软件

***sudo dpkg -r tree*** 卸载软件

注：将tree.deb传到Linux系统中，有多种方式。VMwareTool，使用挂载方式；使用winSCP工具等；

APT（Advanced Packaging Tool）高级软件工具。这种方法适合系统能够连接互联网的情况。依然以tree为例:

***sudo apt-get install tree*** 安装tree

***sudo apt-get remove tree*** 卸载tree

***sudo apt-get update*** 更新软件

***sudo apt-get upgrade***

将.rpm文件转为.deb文件

.rpm为RedHat使用的软件格式。在Ubuntu下不能直接使用，所以需要转换一下。

***sudo alien abc.rpm***

**vim使用**

vim三种模式：命令模式、插入模式、编辑模式。

使用ESC或i或：来切换模式。

命令模式下：

***:q*** 退出

***:w*** 保存

***:wq*** 保存并退出

***:q!/:wq!*** 强制退出

***:set nu*** 显示行号（nu 是简写 全拼为 number）

***:set nonu*** 隐藏行号（同上）

***/apache*** 在文档中查找apache 按n跳到下一个，shift+n上一个

***yyp*** 复制光标所在行，并粘贴

***yy*** 复制

***p*** 粘贴

***h*** (左移一个字符←)

***j*** (下一行↓)

***k*** (上一行↑)

***l*** (右移一个字符→)

**用户及用户组管理**

***/etc/passwd*** 存储用户账号

***/etc/group*** 存储组账号

***/etc/shadow*** 存储用户账号的密码

***/etc/gshadow*** 存储用户组账号的密码

***useradd*** 添加用户名

***userdel*** 删除用户名

***adduser*** 添加用户名

***groupadd*** 添加组名

***groupdel*** 删除组名

***passwd root*** 给root设置密码

***su root*** 变更帐号为root

***su -root*** 变更帐号为root并改变工作目录至root的家目录

***/etc/profile*** 系统环境变量

***bash\_profile*** 用户环境变量

***.bashrc*** 用户环境变量

***su user*** 切换用户，加载配置文件.bashrc

***su -user*** 切换用户，加载配置文件/etc/profile ，加载bash\_profile

**更改文件的用户及用户组**

***sudo chown [-R] owner[:group] {File|Directory}***

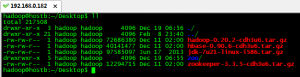
例如：还以jdk-7u21-linux-i586.tar.gz为例。

属于用户hadoop，组hadoop要想切换此文件所属的用户及组。可以使用命令。

***sudo chown root:root jdk-7u21-linux-i586.tar.gz***

**文件权限管理**

*三种基本权限*  
R 读 数值表示为4  
W 写 数值表示为2  
X 可执行 数值表示为1

  
如图所示，jdk-7u21-linux-i586.tar.gz文件的权限为-rw-rw-r–  
-rw-rw-r–一共十个字符，分成四段。  
第一个字符“-”表示普通文件；这个位置还可能会出现“l”链接；“d”表示目录  
第二三四个字符“rw-”表示当前所属用户的权限。 所以用数值表示为4+2=6  
第五六七个字符“rw-”表示当前所属组的权限。 所以用数值表示为4+2=6  
第八九十个字符“r–”表示其他用户权限。 所以用数值表示为4  
所以操作此文件的权限用数值表示为664?

**更改权限**

***sudo chmod [u所属用户 g所属组 o其他用户 a所有用户] [+增加权限 -减少权限] [r w x] 目录名***

例如：有一个文件filename，权限为“-rw-r—-x” ,将权限值改为”-rwxrw-r-x”，用数值表示为765

***sudo chmod u+x g+w o+r filename***

上面的例子可以用数值表示

***sudo chmod 765 filename***