KUAS
KYOTO UNIVERSITY OF ADVANCED SCIENCE
京都発世界人財

京都先端科学大学

Information Literacy 06

command line

Ian Piumarta

Faculty of Engineering, KUAS

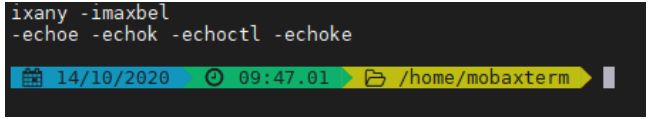
open a command line

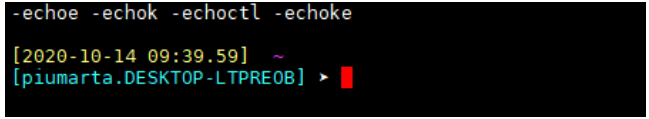
Mac: open  `Terminal.app`

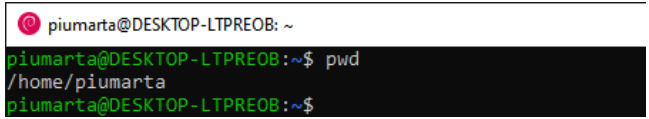
MobaXterm: click  MobaXterm icon

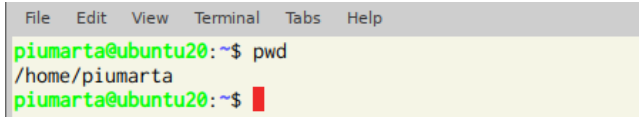
WSL: click  Debian or  Ubuntu icon

the shell waits for you to type a command









the shell executes commands that you type

if you type a command wrong, the shell will tell you

```
$ pwd
/home/piumarta
$ dwp
dwp: command not found
```

if a command is ‘stuck’, type `Control-C` to interrupt it

- hold down `Control` while typing `C`

pwd tells you where you are

the `pwd` command **p**rints the current **w**orking **d**irectory

```
$ pwd
/home/piumarta
```

you are always working in one specific directory

path names are relative to the working directory

- unless they start with '`/`', making them *absolute*
  (i.e., relative to the root directory)

cd changes the current working directory

```
$ cd /home
$ pwd
/home
```

cd without an argument changes to your home directory

```
$ cd
$ pwd
/home/piumarta
```

cd - goes back to the previous directory

```
$ cd -
$ pwd
/home
$ cd -
$ pwd
/home/piumarta
$ cd -
$ pwd
/home
```

`ls` lists information about files and directories

with no argument `ls` lists the current directory

```
$ ls
Desktop  LauncherFolder  MyDocuments
```

with an argument `ls` lists the given directory

```
$ ls /home
mobaxterm  piumarta
```

options change the behaviour of commands

```
$ ls
mobaxterm  piumarta
$ ls -a
.  ..  mobaxterm  piumarta
$ ls -l
total 8
drwxr-xr-x 1 piumarta UsersGrp 0 Oct 18 17:04 mobaxterm
drwxr-xr-x 1 piumarta UsersGrp 0 Oct 18 17:04 piumarta
$ ls -F
mobaxterm/  piumarta/
```

-a = list hidden files (start with '.')

-l = long format (information in inode)

-F = classify (folder/, executable)*

common pattern: *command options arguments*

```
$ ls      -laF    /home /usr
  ↑       ↑       ↑
  command options arguments
```

spaces separate one item from the next

the number of spaces is not important

```
$     ls   -laF     /home     /usr
```

~ stands for your home directory

```
$ ls ~
Desktop  LauncherFolder  MyDocuments
```

you can use it as part of a longer file name

```
$ ls -F ~/..
mobaxterm/  piumarta/
```

`mkdir` creates directories

```
$ cd
$ mkdir temp
$ ls
Desktop  LauncherFolder  MyDocuments  temp
$ cd temp
$ pwd
/home/piumarta/temp
$ ls -la
total 4
drwxr-xr-x 1 piumarta Users 0 Oct 18 21:22 .
drwxr-xr-x 1 piumarta Users 0 Oct 18 21:22 ..
```

Q: 1–4

cp copies files

```
$ pwd
/home/piumarta/temp
$ ls
$ cp /etc/hosts .
$ ls -l
-rw-r--r-- 1 piumarta Users 852 Oct 23 02:04 hosts
```

Q: 5

> redirects output to a file, `cat` concatenates files

```
$ pwd
/home/piumarta/temp
$ ls -la > output
$ cat output
total 4
drwxr-xr-x 1 piumarta Users 0 Oct 22 21:43 .
drwxr-xr-x 1 piumarta Users 0 Oct 22 21:27 ..
-rw-r--r-- 1 piumarta Users 0 Oct 22 21:43 output
```

why is `output` included in the output from `ls`?

why does `output` always appear empty in the listing?

Q: 6

`grep` finds patterns in the contents of files

```
$ grep bash /etc/passwd
root:x:0:0:root:/root:/bin/bash
piumarta:x:501:20:Ian Piumarta:/home/piumarta:/bin/bash
$ ls /bin > list
$ grep bash list
bash
rbash
$ grep bash /etc/passwd list
/etc/passwd:root:x:0:0:root:/root:/bin/bash
/etc/passwd:piumarta:x:...:/home/piumarta:/bin/bash
list:bash
list:rbash
```

commands like `grep` can take input from the keyboard

```
$ grep secret
find
the
secret
secret
word
```

`Control-C` to terminate

Q: 7

< redirects input from a file

```
$ cat > words
find
the
secret
word
$ grep secret < words
secret
```

Control-C to terminate

| pipes command output to another command's input

```
$ ls /bin | grep oo
chroot
fvwm-root.exe
nslookup
regtool.exe
VNCHooks.dll
xpmroot
```

compare: `cmd1 > tmp` followed by `cmd2 < tmp`

with: `cmd1 | cmd2` (shorter, avoids the temporary file)

wc counts characters, words, and lines

```
$ wc /etc/passwd
11    13 1065 /etc/passwd
$ wc
it was the best of times
it was the worst of times
2     12    51
```

lines, words, characters, filename

input from keyboard or pipe

Q: 8–9

a few simple commands make a powerful pipeline

there is no command to count the frequency of words

but you can easily make your own out of smaller commands

```
$ cat words
it was the best of times
it was the worst of times
$ tr ' ' '\012' < words | sort | uniq -c | sort -n
    1 best
    1 worst
    2 it
    2 of
    2 the
    2 times
    2 was
```

tr ' ' '\012' = transform spaces (' ') into newlines ('\012')

sort = sort input lines alphabetically

uniq -c = remove duplicates, print how many times each line is repeated

sort -n = sort input lines numerically

Q: 10

`rm` removes files, `rmdir` directories

```
$ cd
$ rmdir temp
rmdir: temp: Directory not empty
$ rm temp/*
$ rmdir temp
$ ls temp
ls: temp: No such file or directory
```

'' matches any filename*

`rm` removes files, `rmdir` directories

```
$ cd
$ rmdir temp
rmdir: temp: Directory not empty
$ rm temp/*
$ rmdir temp
$ ls temp
ls: temp: No such file or directory
```

'' matches any filename*