

Aliases, functions, wrappers, pipelines

Virginie Orgogozo

April 6 2012

Alias

shortcut for a commonly used shell command

~/.bash_profile or ~/.bashrc file:

```
(...)  
  
alias cdp='cd ~/pcfb/sandbox'  
  
alias cx='chmod u+x'  
  
alias ag='agrep -B -y -d ">" '  
  
alias myjobs="ps -ax | grep orgogozo"  
  
alias hg='history | grep'
```

In the shell:

```
$cdp  
  
$cx script2.py  
  
$ag CYG examples/FPexcerpt.fta  
  
$myjobs  
  
$hg pcfb  
  
$alias (list your defined aliases)
```

For the changes in your .bashrc file to be used in the shell:

- open a new shell window

or: - `$source .bashrc`

Functions

For a more complex set of shell commands

In the shell:

```
$pdftk GenomeBiolPaper.pdf cat 2-end output Doe2012.pdf
```

```
$firstpage GenomeBiolPaper.pdf Doe2012.pdf
```

~/.bash_profile or **~/.bashrc** file:

```
(...)  
  
function firstpage() { pdftk $1 cat 2-end output $2 ;}  
  
listall() {  
  ls -la  
  echo "Above are all the directories of the following  
  folder:"  
  pwd  
  date  
}
```

```
$listall
```

You can also write a function in the shell only

```
virginie@Darwin:~$ repeater(){  
> echo "$1 is what you said first"  
> echo "$@ is everything you said"  
> }
```

Defines
the
function

```
virginie@Darwin:~$ repeater hello everybody! How are  
you?  
hello is what you said first  
hello everybody! How are you? is everything you said  
virginie@Darwin:$
```

Usage of
the
function

Use a function when you want to use the same command multiple times with different filenames in the middle

```
virginie@Darwin:~$ phym1 sequencesA.fta 1 i 1 100 WAG  
0 8 e BIONJ y y
```

```
virginie@Darwin:~$ phym1 sequencesB.fta 1 i 1 100 WAG  
0 8 e BIONJ y y
```

```
virginie@Darwin:~$ phym1 CG5679.fta 1 i 1 100 WAG 0 8  
e BIONJ y y
```

```
virginie@Darwin:~$ phym1 ASCRE.fta 1 i 1 100 WAG 0 8  
e BIONJ y y
```

(...)

```
virginie@Darwin:~$myphym1( ){  
> phym1 $1 1 i 1 100 WAG 0 8 e BIONJ y y  
> }
```

Defines
the
function

```
virginie@Darwin:~$ myphym1 sequencesA.fta
```

```
virginie@Darwin:~$ myphym1 sequencesB.fta && myphym1  
CG5679.fta && myphym1 ASCRE.fta
```

Usage of
the
function

Use a function to rename multiple files

file1.txt	→	u_file1.dat
file2.txt		u_file2.dat
file3.txt		u_file3.dat
...		...

```
renamer() {
    EXT="dat"
    PRE="u_"
    if [ $# -lt 1 ] (if the number of arguments is ..., then...)
    then
        echo "Rename a file.txt list as $PREfile."
    $EXT"
    else
        for FILENAME in "$@"
        do
            ROOTNAME="{FILENAME%.*}"
            cp "$FILENAME" "$PRE$ROOTNAME.$EXT"
            echo "Copying $FILENAME to
$PRE$ROOTNAME.$EXT"
        done
    fi
}
```

Copy and paste this text from `pcfb/scripts/shellfunctions.sh` in your shell

```
virginie@Darwin:~$ cd ~/pcfb/examples/spectra/  
virginie@Darwin:~/pcfb/examples/spectra$ ls  
LEDBlue.txt LEDGreen.txt LEDRed.txt LEDYellow.txt
```

```
virginie@Darwin:~/pcfb/examples/spectra$ renamer *.txt  
Copying LEDBlue.txt to u_LEDBlue.dat  
Copying LEDGreen.txt to u_LEDGreen.dat  
Copying LEDRed.txt to u_LEDRed.dat  
Copying LEDYellow.txt to u_LEDYellow.dat
```

```
virginie@Darwin:~/pcfb/examples/spectra$ ls  
LEDBlue.txt LEDRed.txt u_LEDBlue.dat u_LEDRed.dat  
LEDGreen.txt LEDYellow.txt u_LEDGreen.dat u_LEDYellow.dat  
virginie@Darwin:~/Documents/WWW/BioInfoCourses/pcfb/examples/spectr  
a$
```

Use a loop to repeat operations

```
virginie@Darwin:~$for k in {1..10}
```

```
>do
```

```
>echo $k
```

```
>done
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
8
```

```
9
```

```
10
```

```
virginie@Darwin:~$
```

```
virginie@Darwin:~$for i in {A..Z}; do mkdir $i-  
authors; mv $i*.pdf $i-authors; done
```


Wrappers

Program that controls and expands the functionality of another program

Ex: python script that calls MATLAB or another program

Pipelines

Automated workflow

Very important for efficiency, consistency and record