basic shell commands

previous (notes0906-intro-shell.html) & next (notes0915.html)

the Unix shell

Go to software carpentry introduction (http://swcarpentry.github.io/shell-novice/) and do

- setup (http://swcarpentry.github.io/shell-novice/setup.html) to download the data and open a terminal

After you do the setup and have a terminal open, type `echo $SHELL`. You may see this:

```
$ echo $SHELL
/bin/bash
```

or you may get this:

```
% echo $SHELL
/bin/zsh
```

The terminal is the “window” (more or less), while the shell is a program (or a programming language, like R and Python are). There are several shell programs, `bash` (and `zsh`) being the most common. They are almost equivalent.

bash (https://en.wikipedia.org/wiki/Bash_(Unix_shell)) for “Bourne-Again SHell”, pun on the name of the developer of the original Unix shell, Stephen Bourne.

Then do:

- navigating files and directories (http://swcarpentry.github.io/shell-novice/02-filedir/index.html)
- working with files and directories (http://swcarpentry.github.io/shell-novice/03-create/index.html)
  - do things on your own laptop. watching or reading ≠ practice.
  - ask others on Piazza, or in office hours.

summary

- directory structure, root is `/`
- relative versus absolute paths
  - in your code and projects: use **relative** paths as much as possible: it makes your code more portable, for others, and for yourself if you re-locate your own project folder
- shortcuts: `.`, `..`, `~`, `-`
  - `cd -` is so useful!
- tab completion to get program and file names
- up/down arrows and `!` to repeat commands

`whoami`	who am I? to get your username
`pwd`	print working directory. where am I?
`ls`	list. many options, e.g. `-a` (all) `-l` (long) `-lrt` (reverse-sorted by time)
`cd`	change directory
`mkdir`	make directory
`rm`	remove (forever). `-f` to force, `-i` to ask interactively, `-r` recursively
`rmdir`	remove (delete) directory, if empty
`mv`	move (and rename). can overwrite existing files, unless `-i` to ask
`cp`	copy. would also overwrite existing files
`touch`	create blank file, or modify time stamp of existing file
`diff`	difference
`wc`	word count: lines, words, characters. `-l`, `-w`, `-c`
`cat`	concatenate
`less`	because “less is more”. `q` to quit.
`sort`	`-n` for numerical sorting
`head`	first 10 lines. `-n 3` for first 3 lines (etc.)
`tail`	last 10 lines. `-n 3` for last 3 lines, `-n +30` for line 30 and up
`uniq`	filters out repeated lines (consecutive). `-c` to get counts
`cut`	cut and return column(s). `-d,` to set the comma as field delimiter (tab otherwise), `-f2` to get 2nd field (column)
`echo`	print
`history`	shows the history of all previous commands, numbered
`!`	`!76` to re-execute command number 76 in the history, `!$` for last word or last command

file names

so important: **no spaces!** example:

- create a directory ‘raw sequences’ in `data`, using a GUI (e.g. Finder)
- try to remove it from the command line:

```
cd data
ls
rm -rf raw sequences
```

lucky for us: `raw` or `sequences` didn't exist (chainsaw…)
how can we remove this directory?

- prefer lower-case letters, especially for the first letter of a file name: time saver, along with tab completion
- common usage: capitalize between words, or underscores, or `-`, like `wheatSequenceAlignments` (camel case style) or `wheat_sequence_alignments` (snake case style).
- use ASCII characters only, no space (did I mention this already?), no `/`, no `\` (for Windows), no `-` for the first character.
- R users: avoid dots. conventionally used for the file extension.

Great presentation (https://speakerdeck.com/jennybc/how-to-name-files) by Jenny Bryan

- choose file names to ease automation, using shell expansion
- use leading zeros: `file-0021.txt` rather than `file-21.txt`. lexicographic sorting files (like with `ls`) would otherwise place `file-1390.txt` before `file-21.txt`.
- file extensions:
  - not needed by the computer. for humans only.
  - explicit is better than implicit for humans. ex: `rice_genes.fasta` versus `rice_genes`
  - used by the computer occasionally: to pick the 'default' app to open a file (e.g. `open xxx.pdf` or `xdg-open xxx.pdf`); to color parts of text by a text editor; etc.

typing skills

- had keyboarding classes in elementary school?
- it's like talking or walking: it's assumed.
- take a test (http://www.typingtest.com/test.html)
- invest in your typing skills! it will save you time and stress.
  allow yourself two weeks to be slow.

---

previous (notes0906-intro-shell.html) & next (notes0915.html)
