

Shell: Advanced

Find things

1. `find . -name haiku*`

Look at files

The haiku

1. `cat ./folders/subfolder2/subsubfolder6/haiku.txt`
2. `find . -name "haiku*" -exec cat {} ;`

Our yellow.md

1. `vscode colors/yellow.md`
2. `cat`
3. `head`
4. `tail`

Pipes

How to get only the first real paragraph?

Combine them through **pipes**

1. `head -n 11 colors/yellow.mp3`
2. `head -n 11 colors/yellow.mp3 | tail -n 7`
3. `head -n 11 colors/yellow.mp3 | tail -n 7 > colors/yellow-paragraph.md`

How to sort and filter lines in files?

Note: only deletes sequential lines

1. `cat glaciers.txt`
2. `uniq glaciers.txt`
3. `sort glaciers.txt`
4. `uniq glaciers.txt | sort`
5. `sort glaciers.txt | uniq > glaciers-unique.txt`

How to count lines, words and characters

1. `wc glaciers.txt`
2. `uniq glaciers.txt | wc`
3. `sort glaciers.txt | uniq | wc`

pandoc

1. `pandoc animals/rhino.md -o animals/rhino.pdf`
2. `pandoc animals/rhino.md -o animals/rhino.html`

How to automate things (little shell scripts)

1. code `sort-uniq.sh`
2. `sort "$1.txt" | uniq > "$1-su.txt"`

two inputs version

`sort "$1" | uniq > "$2" && wc "$1" && wc "$2"`

What have we learned?

- **pandoc input -o output:** Easily convert markdown into pdf or html (and many more)
- **bash shell-script arguments:** Execute a shell-script
- **\$N:** Placeholder variable for N^{th} argument

Exercise: Small automatisation

1. Go into the `animals` folder
2. Open a shell-script called `make-pdf.sh` with VS Code
3. We want to automate the pandoc pdf creation from before
 - The script takes one argument, the input filename without the fileending, e.g. `rhino`
 - from `rhino.md` it creates `rhino.pdf`
4. Go into the `animals` folder and try what happens if you type `bash make-pdf.sh rhino`
5. delete `rhino.pdf`
6. What happens if you type `bash make-pdf.sh {rhino, lion}`?
7. Do have an idea what the problem could be? Discuss with your partner.

Rest see solutions