

# DST4L Copenhagen (December 2016), Grappling with Data

---

## Basics - navigating the shell

**pwd** print working directory

**ls** list directory

- **-l**: list file information
- **-lh**: list human readable file information

**cd** change directory

---

## Basics - interacting with files

**mkdir** make directory

**cat** send file or files to output (in most cases, print to shell)

**head** output first parts of a file or files

**tail** output last parts of a file or files

**mv** rename or move a file or files. Syntax for renaming a file: **mv FILENAME NEWFILENAME**

**cp** copy a file or files. Syntax: **cp FILENAME NEWFILENAME**

**>** redirect output. Syntax with **cat**: **cat FILENAME1 FILENAME2 > NEWFILENAME**

**rm** remove a file or files. NB: *USE WITH CAUTION!!!*

---

## Wildcards

**?** a placeholder for one character or number

**\*** a placeholder for zero or more characters or numbers

**[ ]** defines a class of characters

*Examples*

- **foobar?**: matches seven character strings starting with foobar and ending with one character or number
- **foobar\***: matches strings starting with foobar ending with zero or more further characters or numbers
- **foobar\*.txt**: matches strings starting with foobar and ending with .txt
- **[1-9]foobar?**: matches eight character strings starting that start with a number, have foobar after the number, and end with any character or number.

---

## Counting and Mining

**wc** word count

- `-w`: count words
- `-l`: count lines
- `-c`: count characters (or `m` for Mac users)

**grep** global regular expression print

- `-c`: displays counts of matches for each file
- `-i`: match with case insensitivity
- `-w`: match whole words
- `-v`: exclude match
- `--file=FILENAME.txt`: use the file `FILENAME.txt` as the source of strings used in query

**egrep** pattern matching global regular expression print

- `--file=FILENAME.txt`: use the file `FILENAME.txt` as the source of strings used in query, this file may contain a regular expression

---

## References and further reading

- Command Line Crash Course <http://cli.learncodethehardway.org/book/>
- Computer Science 101, Coursera <https://www.coursera.org/course/cs101>
- Library Carpentry <http://librarycarpentry.github.io/>
- Programming for Everybody (Python), Coursera <https://www.coursera.org/course/pythonlearn>
- *The Programming Historian* <http://programminghistorian.org/>
  - James Baker, 'Preserving Your Research Data'
  - Ian Milligan and James Baker, 'Introduction to the Bash Command Line'
  - James Baker and Ian Milligan, 'Counting and mining research data with Unix'
- Deborah S. Ray and Eric J. Ray, *Unix and Linux: visual quickstart guide* (4th edition, 2009)
- Software Carpentry <http://software-carpentry.org/>
- Al Sweigart, *Automate the Boring Stuff* (2015) <https://automatetheboringstuff.com/>
- William Turkel, 'Basic Text Analysis with Command Line Tools in Linux' (15 June 2013) <http://williamjturkel.net/2013/06/15/basic-text-analysis-with-command-line-tools-in-linux/>
- William Turkel, 'Pattern Matching and Permuted Term Indexing with Command Line Tools in Linux' (20 June 2013) <http://williamjturkel.net/2013/06/20/pattern-matching-and-permuted-term-indexing-with-command-line-tools-in-linux/>