regex & grep

Text Matching

- Exact matching: matches a single string
- Pattern matching: can match a set of strings

Regular Expression (regex)

- a sequence of characters that specify a match pattern
- each character in the regex is either a literal character or a metacharacter
    - literal character: exact match
    - metacharacter: a character with a special meaning

Regex Metacharacters

- vertical bar for or: `gray|grey` matches "gray" or "grey"
- parentheses used for scope: `gr(a|e)y` matches "gray" or "grey"
- `.` wildcard matches any character: `a.c` matches "abc", "a1c", etc
- quatification:
  - `?` for 0 or 1: `colou?r` matches "color" and "colour"
  - `*` for 0 or more: `ab*c` matches "ac", "abc", "abbc", "abbbc", etc
  - `+` for 1 or more: `ab+c` matches "abc", "abbc", "abbbc", etc

Regex Examples

- `wom[ae]n`: "woman" or women"
- `prince.*`: all strings starting with prince
- `(love|hate|whatever)`: matches "love", "hate", or "whatever"
- `s[ck]eptic. *`: matches different spellings and endings of sceptic

Where can you use regex?

- on the command line
- programming languages
- editors
- other tools

grep

- command line utility that operates on plain text files
- search file(s) via regex
- returns records that match regex
- originally stood for "global regular expression print"
- first application: Federalist Papers authorship
- history of grep, Brian Kernighan

grep examples

- `grep sql mueller.txt` search for string 'sql'
- `grep -i sql mueller.txt` case insensitive
- `grep -i -B 10 sql -A 10 mueller.txt` matching line and ten lines before and after

redaction count examples

- `grep 'Harm to Ongoing Matter' mueller.pdf`
- `grep -c 'Harm to Ongoing Matter' mueller.txt`
- `grep -Ec 'Harm to Ongoing|HOM'  mueller.txt`
- `grep -c 'Personal Privacy' mueller.txt`