

# Text Processing: Takeaways

by Dataquest Labs, Inc. - All rights reserved © 2019

## Syntax

- Concatenating `filename1` and `filename2` : `cat filename1 filename2` .
- Sorting `filename` :
  - Default: `sort filename`
  - Reverse order: `sort - r filename`
  - Removing duplicates: `sort - u filename`
  - With commas as delimiters: `sort - t"," filename` .
    - Sorting by columns 1, 3, and 5 through 7: `sort - t"," - k1,1 k3,3 k5,7`
    - Sorting by column `2` numerically and in reverse: `sort - t "," - k2,2gr`
- Selecting columns `2` , `3` and `7` through `9` on `filename` with `:` as a separator: `cut -d":" - f2,3,7-9`
- Selecting rows of `filename` matching a pattern:
  - Default: `grep 'pattern' filename`
  - Numbering the lines: `grep - n 'pattern' filename`
  - Reverting the match: `grep - v 'pattern' filename`
  - Ignoring case: `grep - i 'pattern' filename`

## Concepts

- Processing text files is a critical skill.
- Text processing is anything we do with text. Some examples are:
  - Reformatting text
  - Extracting specific parts of the text
  - Modifying the text

- The shell is well built to deal with text files because of its tools are relatively easy to use and very efficient.
- Sorting in the shell depends on locale settings.
- There are different kinds of regular expressions, they're mostly similar.
  - Extended regular expressions are the closest to the ones we learned in Python.

## Resources

- [What does text processing mean?](#)
- [Text Processing Commands](#)
- [Locale Explained](#)



Takeaways by Dataquest Labs, Inc. - All rights reserved © 2019