Topic

Interlude: PA3a - Solutions

These Practice Exercises are meant to help you review for our next IE.

Timestamps

- We want to write a command line that allows us to append information about the current time/date into a text file.
- We are going to use the `date` tool, and redirections, to append the current date and time into a text file named *`timestamps.log`* which we will assume is in the working directory.
- We will not be assuming that the file exists, but we will make sure that if we use our solution multiple times, then new lines are appended at the end of the file without overwriting previous information.

```
date >> timestamps.log
```

Improved Timestamps

- Let us get fancy with the format with which the information will be recorded on each line of our text file; Instead of just having the date, we need to have the following information on each line;

  YEAR MONTH DATE DAY HH:MM:SS

- An example of an entry would be;

  2011 February 18 Friday 19:42:00

Hint: *man date* is your friend

```
date +"%Y %B %d %A %H:%M:%S" >> timestamps.log
```

Even Better Timestamps

- What about improving our previous command line so that we are able to also append the user’s name at the end of the line. The previous entry would now read instead;

```
2011 February 18 Friday 19:42:00 smith
```

```
echo `date +"%Y %B %d %A %H:%M:%S"` $USER >> timestamps.log
```

Filtering by Users

- Use the previous command line several times so that you get a timestamps.log file with a variety of entries (or use a text editor to make entries up, even easier).
  - Feed this log file to 1 or more filters so that you only display the username part of each line.
  - Ensure that the list of users is actually sorted
  - Ensure that each username is displayed only once
  - Use a filter to count how many users were in the above list

```
cut -f6 -d ' ' timestamps.log
cut -f6 -d ' ' timestamps.log | sort
cut -f6 -d ' ' timestamps.log | sort | uniq
cut -f6 -d ' ' timestamps.log | sort | uniq | wc -l
```

Filtering by Dates

- We want to feed the timesstamps.log file through one or more filters so that we end up displaying the list of all months, in alphabetical order, that had entries in the file.
- Each month should be displayed only once.

```
cut -f2 -d ' ' timestamps.log | sort | uniq
```

Joining timestamps w/ usernames

We want to use *join* to combine our *timestamps.log* file

```
2011 February 18 Friday 19:42:00 smith
```

With *usernames.log* that lists usernames, followed by a space, and then the full name.

```
mulder Fox Mulder
scully Dana Scully
murph Murphy Cooper
dale Dale Cooper
```

Start by sorting both files based on the usernames and thus create two new versions named *timestamps-sorted.log* and *usernames-sorted.log*.

- Once you have these, use them in your *join* operation and dump the result in a file named *result-sorted.log*.

- The end result of our combination should show all lines from *timestamps.log* with, at the end, the full name of the user (when available). Please note the following;
  - The order will not be chronologic anymore
  - *join* will put the username first on each line
  - Neither of these is an issue.

```
sort -k 6 timestamps.log > timestamps-sorted.log
sort -k 1 usernames.log > usernames-sorted.log
join -a1 -1 6 -2 1 timestamps-sorted.log usernames-sorted.log >
result-sorted.log
```