Topic

Interlude: PA3b - Solutions

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

Reminder: our data files: grades.txt & students.txt

- grades.txt

```
357   90    90    100   95    0
666   42    60    30    90    42
956   42    99    100   75    59
975   89    69    60    30    0
```

- students.txt

```
357   James      McPhearson
555   Alan       Turing
639   Severus    Snape
956   Fox        Mulder
975   Thomas     Anderson
```

Question 1 – Report.1

- At the end of the semester, I want to generate my *report.1* which needs to hold the data illustrated in the following sample;
- I want all students' names to show up, even those for whom I didn't record a single grade

```
357   James      McPhearson      90    90    100   95    0
555   Alan       Turing
639   Severus    Snape
956   Fox        Mulder          42    99    100   75    59
975   Thomas     Anderson        89    69    60    30    0
```

```
join -a1 students.txt grades.txt > report.1
```

Question 2 – Report.2

- At the end of the semester, I want to generate my *report.2* which needs to hold the data illustrated in the following sample;
- I don’t want to see in this report the names of students without any grade. Provide the shell line you issued to achieve this goal.

```
357  James McPhearson    90   90   100  95   0
956  Fox Mulder          42   99   100  75   59
975  Thomas Anderson     89   69   60   30   0
```

```
join students.txt grades.txt > report.2
```

Question 3 – Report.3

- I'd like to make sure I don't have any lines in *grades.txt* which correspond to a student who doesn't have an entry in *students.txt*.
- To determine this, I usually generate a *report.3* file which holds the lines from *grades.txt* with a student ID not present in any entry of *students.txt*.
- The following would be the data in our *report.3* based on what we have so far in our example data files;

```
666   42    60    30    90    42
```

```
join -v2 students.txt grades.txt > report.3
```

Question 4 – Report.4

- I’d like to generate a file named *report.4* holding the list of all students who didn’t receive any grades. This means students with an entry in *students.txt* but without one in *grades.txt*.
- Based on our example data files, here’s an example of what *report.4* would hold after using your solution;

```
555  Alan      Turing
639  Severus   Snape
```

```
join -v1 students.txt grades.txt > report.4
```

Question 5 – Report.5

- Using students IDs is fine for these intermediary reports but, at the end of the semester, I want to generate my *report.5* which needs to look like the following sample;
- Only students with grades are in this report. For each of them, I have their first name, last name, and then all their grades. Provide the shell line you issued to generate this file

```
James McPhearson    90   90   100  95   0
Fox Mulder          42   99   100  75   59
Thomas Anderson     89   69   60   30   0
```

```
join students.txt grades.txt| colrm 1 4 > report.5
```

Question 6 – Report.5 Improved

- Assuming I have my *report.5* file, how do I order it alphabetically by first names? When I’m done, I want the same *report.5* file to be ordered as required instead of having to generate yet another text file. Here’s an example of applying this solution to the above *report.5* file;

```
Fox Mulder              42    99    100   75    59
James McPhearson        90    90    100   95    0
Thomas Anderson         89    69    60    30    0
```

```
sort -o report.5 report.5
```

Question 7 – Report.5 Rounded up

- This is looking better but now I’d like to modify my *report.5* file to replace all occurrences of a 69 by 70, 79 by 80, 89 by 90 and 99 by 100. I still want to do so with a single, even if rather long command. As with question #6, I don’t want to generate yet another text file but instead modify directly *report.5*. Here’s an example of the result of applying this solution to *report.5*;

```
Fox Mulder                 42    100   100   75    59
James McPhearson           90    90    100   95    0
Thomas Anderson            90    70    60    30    0
```

```
sed -i -e ‘s/69/70/g’ -e ‘s/79/80/g’ -e ‘s/89/90/g’ -e
‘s/99/100/g’ report.5
```

Question 8 – Report.5 top students

- Now, I'd like to be able to extract, from the previous *report.5* file, the first and last names of all the students who have at least a grade of 100 in their list of grades. This list will have to be dumped into another text file named *top.txt*.
- You will have to use both the *grep* and *cut* tools in your solution so read their respective manpages. Here's an example of the resulting *top.txt*;

```
Fox Mulder
James McPhearson
```

```
grep 100 report.5| cut -f 1-2 -d ‘ ‘ > top.txt
```

Question 9 – Report.5 Reverse 1st / last name order

- I want to reverse the order of the first and last names in each line of the *report.5* file we've been working with. Then, I'll sort it again by last name this time.
- Note that I don't want to generate more files, just operate on the one I already have from the previous questions.
- You might have to use several tools, and produce a few temporary files (remember to erase them when you are done).
- Here's an example of *report.5* after applying this solution;

```
Anderson Thomas90   70   60   30   0
McPhearson James    90   90   100  95   0
Mulder Fox     42   100  100  75   59
```

Question 9 – Report.5 Reverse 1st / last name order

```
cut -f 1-d ‘ ‘ report.5 > firstnames
cut -f 2 -d ‘ ‘ report.5 > lastnames
cut --complement -f1,2 -d ‘ ‘ report.5 > allgrades
paste lastname firstname allgrades|sort -o report.5
rm firstnames lastnames allgrades
```

Question 10 – one last sorting

- Just out of boredom, I’m wondering how I would have been able to sort the *report.5* from question #7 by last name without swapping around the first and last names and by only using the *sort* tool.
- The result would still be to modify *report.5* instead of generating yet another text file. Here’s an example of what the result would look like;

```
Thomas Anderson90   70   60   30   0
James McPhearson    90   90   100  95   0
Fox Mulder          42   100  100  75   59
```

```
sort -k 2 -o report.5 report.5
```