

Week report 7 - sed & awk

For each of the commands in the presentation, include the following:

Command	What it does	Syntax	Example	Example 2
cat	displaying the content inside of a file	<code>cat + option + file</code>	<code>cat -n ~/Downloads/games.txt</code>	<code>cat -b ~/Downloads/video.md</code>
tac	displays content in file in reverse	<code>tac + option + file</code>	<code>tac -n ~/Documents/todo.md</code>	<code>tac -E ~/Document/todo.md</code>
head	Display first 10 lines of an file (by default) you can specify	<code>head + option + file</code>	<code>head -5 /Top5/games</code>	<code>head /top10/Movies</code>
tail	Display first 10 lines of an file (by default) you can specify in reverse	<code>tail + option + file</code>	<code>tail -7 top7/cars</code>	<code>tail top10/food</code>
cut	extracts a specific section of each line of a file and displays it on screen	<code>cut + option + file</code>	<code>cut -b 1-8 students.txt</code>	<code>cut -d ":" -f1,7 --outputdelimiter' => ' /etc/passwd</code>
paste	Used to join to files together horizontally	<code>paste + option + file</code>	<code>paste games.txt movies.txt</code>	<code>paste -d ":" users.lst ipAddress.lst</code>

Command	What it does	Syntax	Example	Example 2
sort	Sorts displayed file in alphabetical order or reverse order, whatever you want	<code>sort + option + file</code>	<code>sort passwd.txt</code>	<code>sort -r users.txt</code>
wc	used for printing number of lines character or bytes in a file (depending on option)	<code>wc + option + file</code>	<code>wc -m random.txt</code>	<code>wc -l users.txt</code>
tr	translates a specific character in a file from one to another	<code>standard output tr + option + set + set</code>	<code>cat file.txt tr '.' '.'</code>	<code>cat program.py tr "[:space:]" '\t'</code>
diff	compare files and display difference	<code>diff + option + file1 + file2</code>	<code>diff playstation.txt xbox.txt</code>	<code>diff -y books.csv magazine.csv</code>
grep	is used to find given text inside of file (like using ctrl + f)	<code>grep + option + file + search criteria + file(s)</code>	<code>grep 'Mom' ~/History/familyTree.csv</code>	<code>grep -n 'Halo' games.txt</code>

Command	What it does	Syntax	Example	Example 2
awk	used for processing and displaying text. Preforms operation line by line.	awk + options + {awk command} +file + file to save (This is optional)	awk -F '{print toupper(\$)}' /passwd	awk '{print \$1}' ~/Documents/cars.csv
sed	Runs operations on a file, it can search, insert, find, replace, and delete.	sed + option + sed script + file	sed 's/re4/Bloodborne' Bestgame.csv	sed '3 s/glue/gorillaGlue/ topGlue.lst

For *sed* and *awk*, please include at least 5 examples

awk examples

- 1 Print the first and last field of "games.csv"

```
awk -F: '{print $1," = ", $NF}' ~/List/games.csv
```

- 2 Print first field of movies.txt

```
awk -F: '{print $1}' ~/List/games.csv
```

- 3 Print the first and 4th field with a different field separator

```
awk -F: '{OFS="="}{print $1,$4}' ~/Documents
```

- 4 print the last field of the movie file

```
awk -F: '{print $NF}' ~/List/movie.csv
```

- **5 Print first field and 3 field with line numbers**

```
awk -F: '{print NR,$1,$3}' ~/Classroom
```

sed examples

- **1 Replacing string on a range of lines**

```
sed '1,3 /CODMW3/BaulderGate3' GameOfTheYear.csv
```

- **2 Replacing string on a specific line number**

```
sed '6 /BloodBorne/DemonSouls' GameOfTheYear.csv
```

- **3 To delete the last line**

```
sed '$d' GameOfTheYear.csv
```

- **4 To insert two blank lines**

```
sed 'G;G' GameOfTheYear.csv
```

- **5 To delete particular line(line 8)**

```
sed '8d' GameOfTheYear.csv
```

Bryant F. Polanco