LINUX FOUNDATION

TRAINING

Manipulating Text Files on the Command Line

A Linux Foundation Training Publication
www.training.linuxfoundation.org

Overview

You can perform some basic operations on a text file without opening a text editors. In some cases, it is more efficient to use command line text manipulation tools than manually performing the same operations using text editors.

Key Ideas

Standard output (stdout): Most commands direct their results to the screen. This is known as standard output, or stdout.

cat: The cat command is short for catenate, and displays the contents of a file on screen.
Usage: cat filename.txt

tail: The tail command displays the last 10 lines of a file on screen. Press “q” to exit.
Usage: tail filename.txt

head: The head command displays the first 10 lines of a file on screen. Press “q” to exit.
Usage: head filename.txt

> or >>: Used to direct the output of a command to somewhere other than the screen. >> appends the output to a destination location (like a file), whereas > overwrites any existing content at the destination location.
Usage: cat filename.txt > newfile.txt

echo: The echo command repeats any text that comes after it back to the screen.
Usage: echo hello world

paste: The paste command puts the lines from two files together, side by side (by default) or serially.
Usage: paste file1.txt file2.txt

sort: The sort command sorts the contents of a file, alphabetically by default.
Usage: sort file1.txt

sed: The sed command is used to search for and replace characters.
Usage: sed ‘s/term/replacement/flag’ file where “term” is the character(s) you are replacing, and “replacement” is the character(s) you are replacing it with. If no flag is given, the first instance of “term” is replaced, if the “g” flag is given all instances are replaced.

Example Scenario

Create 2 text files called fruit.txt and colour.txt. On separate lines in fruit.txt, enter “apple, grape, banana”. On separate lines in colour.txt, enter “red, purple, yellow”. Now you have 2 text files with 3 lines each.

Now Do It

Practice manipulating fruit.txt and colour.txt to achieve the following results.

1. Display the contents of both fruit.txt and colour.txt one after the other.
2. Display the contents of both fruit.txt and colour.txt side by side
3. Add a line with the word “watermelon” to fruit.txt and a line with the word “pink” to fruit.txt
4. Create a new file called fruitcolour.txt that contains the content of fruit.txt and colour.txt side by side.
5. Create a new file called reversefruitcolour.txt that contains the content of fruit.txt and colour.txt side by side sorted in reverse alphabetical order.
6. Change reversefruitcolour.txt so that it only contains the line “I am colour blind”.
7. Take reversefruitcolour.txt, replace “blind” with “coordinated”, and output to snappydresser.txt

If you remember nothing else...

Most text manipulation commands return their results to the screen (stdout). You can return those results to a file instead by redirecting the output using > or >>.

Answer Key

1. Display contents consecutively

```
# cat fruit.txt colour.txt
```

2. Display contents side by side

```
# paste fruit.txt colour.txt
```

3. Add a line to each text file

```
# echo “watermelon” >> fruit.txt
# echo “pink” >> colour.txt
```

4. Create a new file with the contents of each side by side

```
# paste fruit.txt colour.txt > fruitcolour.txt
```

5. Create a new file with the contents in reverse alpha order

```
# sort -r fruitcolour.txt > reversefruitcolour.txt
```

6. Overwrite the contents

```
# echo “I am colour blind” > reversefruitcolour.txt
```

7. Replace a word in the sentence and write a new file

```
# sed s/blind/coordinated/ reversefruitcolour.txt > snappydresser.txt
```

LINUX FOUNDATION

Get Certified!

Get more information at http://training.linuxfoundation.org/certification/lfcs