In this section we'll discuss moving, renaming, copying, and deleting files and folders. First let's revisit the contents of our current working directory:

```
ls

Code
Documents
Photos
Desktop
Music
echo-out.txt
journal-2017-01-24.txt
todo.txt
```

It's gotten a little sloppy, so let's clean this directory up. First I want to make a new directory to store all of my journal entries in called Journal. We already know how to do that:

```
mkdir Journal
```

Now I want to move my journal entry journal-2017-01-24.txt into the Journal directory. We can **m**o**v**e it using the mv command. mv takes two arguments: first the path to the file or folder that you wish to move followed by the destination folder. Let's try using mv now:

```
mv journal-2017-01-24.txt Journal
ls

Code
Documents
Journal
Photos
Desktop
Music
echo-out.txt
todo.txt

```

Looks like it worked! I just realized however that I want to move the Journal directory into the Documents folder. Thankfully we can do this with mv in the same way:

```
mv Journal Documents
ls

Code
Documents
Photos
Desktop
Music
echo-out.txt
todo.txt
```

Let's just make sure it ended up in the right place:

```
ls Documents

Journal
a-tale-of-two-cities.txt
```

Looks good! Another hidden use of the mv command is that you can use it to rename files and folders. The first argument is the path to the folder or file that you want to rename, and the second argument is a path with the new name for the file or folder. Let's rename todo.txt so it includes today's date:

```
mv todo.txt todo-2017-01-24.txt
ls

Code
Documents
Photos
Desktop
Music
echo-out.txt
todo-2017-01-24.txt
```

Looks like it worked nicely. Similar to the mv command, the cp command **c**o**p**ies a file or folder from one location to another. As you can see cp is used exactly like mv when copying files, the file or folder you wish to copy is the first argument, followed by the path to the folder where you want the copy to be made:

```
cp echo-out.txt Desktop
ls

Code
Documents
Photos
Desktop
Music
echo-out.txt
todo-2017-01-24.txt

ls Desktop

echo-out.txt
```

Be aware that there is one difference between copying files and folders, when copying folders you need to specify the -r option, which is short for *recursive*. This ensures that the underlying directory structure of the directory you wish to copy remains intact. Let's try copying my Documentsdirectory into the Desktop directory:

```
cp -r Documents Desktop
ls Desktop

Documents
echo-out.txt
```

Finally, let's discuss how to delete files and folders with the command line. **A word of extreme caution:** in general I don't recommend deleting files or folders on the command line because as we've discussed before there is ***no undo button*** on the command line. If you delete a file that is critical to your computer functioning you may cause irreparable damage. I *highly* recommend moving files or folders to a designated trash folder and then deleting them the way you would normally delete files and folders outside of the command line (The path to the Trash Bin is ~/.Trashon Mac and ~/.local/share/Trash on Ubuntu). If you decide to delete a file or folder on your computer make absolutely sure that the command you've typed is correct before you press Enter. If you do delete a file or folder by accident stop using your computer immediately and consult with a computer professional or your IT department so they can try to recover the file.

Now that you've been warned, let's discuss rm, the Avada Kedavra of command line programs. When **r**e**m**oving files rm only requires the path to a file in order to delete it. Let's test its destructive power on echo-out.txt:

```
rm echo-out.txt
ls

Code
Documents
Photos
Desktop
Music
todo-2017-01-24.txt
```

I felt a great disturbance in the Force, as if millions of voices suddenly cried out in terror, and were suddenly silenced. - Obi-wan Kenobi

The file echo-out.txt is gone forever. Remember when we copied the entire Documents directory into Desktop? Let's get rid of that directory now. Just like when we were using cp the rm command requires you to use the -r option when deleting entire directories. Let's test this battle station:

```
ls Desktop

Documents
echo-out.txt

rm -r Desktop/Documents
ls Desktop

echo-out.txt
```

Now that the awesome destructive power of rm is on your side, you've learned the basics of the command line! See you in the next chapter for a discussion of more advanced command line topics.

Summary

- mv can be used for moving or renaming files or folders.

- cp can copy files or folders.
- You should try to avoid using rm which permanently removes files or folders.

Mark as completed