Python 1 – I/O

Take a look at file1.txt in a text editor. In the interpreter, make a variable called file1 that contains an open file *buffer* corresponding to file1.txt. Give yourself "read" permissions only.

Type file1.read() and execute. What *type* is the output? Do you notice any characters there that you didn't notice in your text editor (you should)? What are they? Type file1.read() again and check the output. There isn't any! The file buffer type has a position, and .read() will read to the end of the file by default, leaving the position at the end. So when you call it again, it starts from the bottom.

Close the file and open it again, re-assigning to file1. The variable file1 is now "re-bound" to a file buffer whose position is at the top. Type file1.readlines(). What type is the output? The .readlines() method will also leave you at the bottom of the buffer, so you'll have to close and reopen the file again (don't forget those up arrows!). Looping over the file also leaves you at the bottom:

```
for line in file1:
        print line
```

Surely there must be a better way? This time when you reopen the file, capture the output of .readlines() to a variable called, say, "filelist." Now we have the full contents of the file in a *data structure* rather than a buffer object. (Can you figure out how to create an identical variable using a for loop and list method?).

You can loop over this *collection* as many times as you want without having to re-assign the variable. Try it, using a for loop. You can go ahead and close file1 for good, because you now have all its contents stored in memory.

Open a new file buffer object, this time with writing permissions. Call it 'outfile' or something like that. Try writing the contents of "filelist" to "outfile" using the .write() method.

What happened? Your variable filelist is an object of *type 'list'* but python can only write characters to text files. What *type* contains characters? Change filelist into this type, and write it to "outfile." Close "outfile" and take a look at it. Not pretty? Try using the .write() method iteratively with a for loop.