Exercise 7: Reading and writing files

By: Rahul Sharma 24 April, 2018

All this time we were handling the text, numbers and whatever we were printing and taking inputs from the user inside the RAM. If you happen to recollect, a computer has two types of memories: primary and secondary. The secondary memory is the one which holds all our data we "save" to the drive. Since the size of the RAM is limited we require secondary memory to store the information we might need later in time. In this lecture we will try to read and write some secondary memory files, namely, text files and learn how can we store information permanently on the drive and read from it.

This exercise involves writing two files. One is your usual **ex7_1.py** file that you will run, but the other is named **ex7_1_sample.txt**. This second file isn't a script but a plain text file we'll be reading in our script. Here are the contents of that file:

```
This is stuff I typed into a file.
It is really cool stuff.
Lots and lots of fun to have in here.
```

We will use the previous lectures' learnings of `raw_input()` and `argv`. What we want to do is "open" a file and read contents of the file. So, we have to tell Python which file it has to open, secondly, we need to open that file and finally we need to display the contents of the file. In this case the contents are the three lines before this paragraph. Let's understand by doing.

Exercise 1: Reading and displaying the content of text files

1. Create a file called ex7_1.py and write the following code in it.

```
from sys import argv
script, filename = argv
txt = open(filename)
print "Here's your file %r:" % filename
print txt.read()
print "Type the filename again:"
file_again = raw_input("> ")
txt_again = open(file_again)
print txt_again.read()
```

A little explanation: just like raw_input("something") we are using the open("something") "function" here to open the filename you passed in the argv while running the program. Now that you've opened the file in the variable txt, this variable has its different properties, one of which is to let us read the contents of the file. The txt.read() function takes care of reading the contents of the file for us.

2. Above each line, write out in English what that line does.
3. Have your script also do a close() on the txt and txt_again variables. It's important to close files when you are done with them.

Now that we have read the file, that's not the only thing that we would like to do when working with files. We would also like to write to the files and save them for later use. In the second exercise we will be reading the file first, write to it. Before going to the exercise, I'd like you to remember these functions that you can perform on files.

- **close**—Closes the file. Like File- >Save.. in your editor.
- **read**—Reads the contents of the file. You can assign the result to a variable.
- **readline**—Reads just one line of a text file.
- **truncate**—Empties the file. Watch out if you care about the file.
- **write(stuff)**—Writes stuff to the file.

Exercise 2: Reading and writing to a file

1. Create a file called **ex7_2.py** and write the following code in it:

```
from sys import argv
script, filename = argv
print "We're going to erase %r." % filename
print "If you don't want that, hit CTRL- C (^C)."
print "If you do want that, hit RETURN."
raw_input("?")
print "Opening the file..."
target = open(filename, 'w')
print "Truncating the file. Goodbye!"
target.truncate()
print "Now I'm going to ask you for three lines."
line1 = raw_input("line 1: ")
line2 = raw_input("line 2: ")
line3 = raw_input("line 3: ")
print "I'm going to write these to the file."
target.write(line1)
target.write("\n")
target.write(line2)
target.write("\n")
target.write(line3)
target.write("\n")
print "And finally, we close it."
target.close()
```

2. Write a comment above each line telling what is happening in the line.
3. Write a script similar to the last exercise that uses read and argv to read the file you just created. Name the python script **ex7_2_reader.py**
4. There's too much repetition in this file. Use strings, formats, and escapes to print out line1, line2, and line3 with just one target.write() command instead of six.
5. Find out why we had to pass a 'w' as an extra parameter to open. Hint: open tries to be safe by making you explicitly say you want to write a file.