Feb 24th, 2017 (Week 5)
Kelly Ryu

**Today: Advanced File I/O and assignment review**

## I. Advanced I/O

### A. Input

```
variable = open('text_name','r') ## to open file object
var2 = variable.read() # store entire text of file in str var2
var3 = variable.readline() #store next line in var3
```

** remember to take care of newline characters ("\n" for Mac & Linux, "\r\n" for Windows) before manipulating strings(use rstrip method)

e.g.

```
in_file = open('example.txt','r')
s=in_file.read()
print type(in_file)
print type(s)
```

```
→ Output: file
          str
```

e.g. of using `in` operator on files to iterate through lines

```
for line in in_file:
    print line
```

### B. Output

```
var = open('nameOfNewFile.txt','w')
var.write('stuff to write')
var.close() # file only begins to be written when you close it
             # before that, only stored in memory
```

for eg.

```
fred = open('out_file.txt', 'w')
fred.write('very cool')
fred.write('write more stuff')
fred.close()
in_file.close() # also close the read file to conserve memory
```

for e.g. to copy the entire text into another file

```
## this program copies the file example-txt

file_in = open ('example.txt', 'r')
file_out = open ('copy.txt', 'w')

#do the copying
for line in file_in:
    file_out.write(line) #line already has the newline character

file_in.close()
file_out.close()
```

Things to note:

- If a file with the same already exists in directory, newly written file overrides it
- Always remember to close the file after writing AND reading! Program doesn't print in the text file until you close it
- Writing is a time-expensive operation so it does it all at once after you close it
- Consecutive write methods will lead to strings printed continuously on one line
- You need to add `\n` at the end of each line when using the write method to change lines
- 'a' stands for append, or writing at the end of an existing file instead of replacing it as we did with 'w'

### C. Deleting and renaming files (os module)

```
import os
os.remove(textfileName)
os.rename(originalFile, newFileName)
```