

## Lecture 7 Practice Exercises

```
__author__ = 'Noha'
```

**Exercise1: Write a function that counts the number of lines in the input file**

```
## This function counts the total number of line in the input filename
def count_lines(filename, mode):#default mode is 'r'
```

```
#####Testing#####
```

```
count_lines('mbox-short.txt', 'r')
```

```
count_lines ('stuff.txt', 'r')
```

**Exercise 2: Write a function that read the contents of the "mbox-short.txt" file and prints its content without any extra lines**

```
def file_print():
```

```
#####Testing#####
```

```
file_print()
```

**Exercise 3: Write a function that read the contents of the "input1.txt" file and prints its content**

```
## This function reads the contents of file input "input1.txt" and prints the contents
to the output
```

```
def file_read_print(in_filename):
```

```
#####Testing#####
```

```
file_read_print ('input1.txt')
```

**Exercise 4: Write/modify a function to print only the first line of the file "input1.txt" and close it afterwards**

```
## This function prints only the first line of file input1.txt and close the file
def file_print_firstLine(in_filename):
```

```
#####Testing#####
```

```
file_print_firstLine('input1.txt')
```

**Exercise 5: Write a function that read the contents of the "input.txt" file and write its content to the file "output\_EX5.txt"**

```
## This function reads the contents of file input and write the contents to file
output
```

```
def file_read_write(in_filename, out_filename):
```

```
#####Testing#####
```

```
file_read_write ('input.txt', 'output_Ex5.txt')
```

**Exercise 6: Write a function that writes the list Zoo to a file named "output-zoo.txt"**

```
def file_write_list():
```

```
#####Testing#####
```

```
file_write_list ()
```

**Exercise 5: Write a function that function prompts for a file name, read through the file, and look for lines of the form: X-DSPAM-Confidence: 0.8475. When you encounter a line that starts with X-DSPAM-Confidence:**

1. pull apart the line to extract the floating-point number on the line
2. Count these lines and then compute the total of the spam confidence values from these lines.
3. When you reach the end of the file, print out the average spam confidence.

**Test your file on the mbox.txt and mbox-short.txt files.**

Enter the file name: mbox.txt

Average spam confidence: 0.894128046745

Enter the file name: mbox-short.txt

Average spam confidence: 0.750718518519

```
def file_calculate_average(filename_input):
```

```
#####Main program#####
```

```
file_calculate_average('mbox-short.txt')
```

```
file_calculate_average('mbox.txt')
```