**Level 1: Reading a Text File**

1. Open a new Python Repl and run the following program.

fileHandle = open("myfile.txt","r")

fileContents = fileHandle.read()

print(fileContents)

fileHandle.close()

Why does this program produce a run-time error?

*Because the file that is being open does not exist.*

1. Add a text file to your project as follows:
   * Click on “Add File” icon in the files pane/window.
   * Type “myfile.txt” and return.
   * “myfile.txt” is now open in the editor pane/window.
   * Type some text into “myfile.txt”
   * Make sure to add several lines of text

1. Switch back to main.py pile and run the program.
   1. What gets printed out?

*The text that was printed inside the text file was printed on the python output.*

* 1. Explain the result.

1. fileHandle = open("myfile.txt","r")
2. fileContents = fileHandle.read()
3. print(fileContents)
4. fileHandle.close()

*first one says ‘fileHandle’, which means to toggle the file and after it tells it to open, meaning to analyse the file.*

*2nd line tells python to read the contents of the file.*

*3rd line is simple: it tells python to print the contents of the file.*

*Last, finishes the file reading and programming and simply ends the action.*

Load and run the following program.

fileHandle = open("myfile.txt","r")

line = fileHandle.readline()

count = 1

while line :

print("Line ", count, " : ",line.strip())

line = fileHandle.readline()

count += 1

fileHandle.close()

Compare and contrast the output of the first and second program

* 1. How is the read() function similar to the readline() function?

*They both read texts.*

* 1. How is the read() function different from the readline() function?

*Read tells python to read the whole text file, and readline is more specific; to read a specific line.*

Research the Python open() function for file I/O (input / output).

* 1. How do you specify which file to open?

*For example: readline(2) – read line 2.*

* 1. Modify the program to open a different file.

Research how to open a file in a sub-directory.

* 1. Modify the second program to open a file in a sub-directory.

1. import os
2. for root, dirs, files in os.walk('G:\documents'):
3. for file in files:
4. if file.endswith('.txt'):
5. print (os.path.join(root, file))
   1. Demo your program to Mr. Nestor

*complete*

* 1. List your program modifications below

1. import os
2. for root, dirs, files in os.walk('G:\documents'):
3. for file in files:
4. if file.endswith('.txt'):
5. print (os.path.join(root, file))

**Level 2: Writing a Text File**

1. Research the Python open() function for file I/O (input / output).
   1. What does the file mode “r” mean?
   2. What mode is used to open a file for writing?
   3. What other file modes can be used? List and explain their meanings.
2. Load and run the following program.

print("Enter test to write to a file")

print("Type STOP to end the program")

print(" ")

lineNumber = 0

while True :

lineNumber += 1

userPrompt = "Enter Line " + str(lineNumber) + " : "

userText = input(userPrompt)

if userText == "STOP" :

break

print(userText)

1. Modify the program to open a text file for writing.
   1. Demo your program to Mr. Nestor
   2. List your program modifications below
2. Replace the line “print(userText)” with a command to write the value of “userText” to an open file.
   1. Verify that text was written to your file
   2. Demo your program to Mr. Nestor
   3. List your program modifications below

**Level 3: Binary Files**

t.b.d.

<http://www.ece.ualberta.ca/~elliott/ee552/studentAppNotes/2003_w/misc/bmp_file_format/bmp_file_format.htm>