**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()

1. Why does this program produce a run-time error?

This program produces a run-time error because there is no file named “myfile.txt”

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 inside the text file gets printed out.

* 1. Explain the result.

This python program opens a file in read mode, reads the contents of it, prints the contents and closes the file so other programs can make changes to it.

1. 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()

1. Compare and contrast the output of the first and second program
   1. How is the read() function similar to the readline() function?

The read() function is similar to the readline() function because the both return text from the opened file.

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

The read() function is different from the readline() function because the read() function return all the text inside the file whereas the readline() function returns one line of the text and if called again, reads the next line of the text and so on.

1. Research the Python open() function for file I/O (input / output).
   1. How do you specify which file to open?

The first parameter in the brackets of the open() function is used to tell python where to get the file from, or what is the absolute or relative path to the file that you want to open. You can specify which file to open by typing its name in the open() function.

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

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

can be changed to

fileHandle = open("english\_essay","r")

and this will open a file named “english\_essay” if it exists,

1. Research how to open a file in a sub-directory.
   1. Modify the second program to open a file in a sub-directory.
   2. Demo your program to Mr. Nestor
   3. List your program modifications below

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

fileHandle = open("documents/essay.txt","r")

line = fileHandle.readline()

count = 1

while line :

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

line = fileHandle.readline()

count += 1

fileHandle.close()

**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?

The ‘r’ specifies the open function to open the file in read only mode.

* 1. What mode is used to open a file for writing?

There are two different ways to open a file for writing to a file:

* + - Using ‘w’ to clear the file and write to it.
    - Using ‘a’ to add to the already existing file.
  1. What other file modes can be used? List and explain their meanings.

The other modes are

* + - Using ‘x’ to create a file if it does not exist.
    - Using ‘b’ for opening the file in binary mode.
    - Using ‘t’ for opening the file in text mode (same as read)
    - Using ‘+’ to update a file on the disk.

1. 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

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

print("Type STOP to end the program")

print(" ")

filename = 'myfile.txt'

lineNumber = 0

final\_text = ''

**with** open(filename,'a') **as** file:

**while** **True** :

lineNumber += 1

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

userText = input(userPrompt)

**if** userText == "STOP" :

**break**

final\_text = final\_text + '**\n**' + userText

print(userText)

file.close()

print(f"Wrote to the file '*{filename}*': **\n\n**"+final\_text)

1. 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

valid\_inputs = ['append','write']

**def** mode\_select():

is\_valid\_input = **False**

**while** **not** is\_valid\_input:

**try**:

input\_from\_player = (input('Do you want to append or write this file?: '))

**if** input\_from\_player **not** **in** valid\_inputs:

**raise** ValueError *#this will send it to the print message and back to the input option*

**else**:

is\_valid\_input = **True**

**except** ValueError:

print("Invalid choice. Type either 'write' or 'append'")

**else**:

**return** input\_from\_player

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

print("Type STOP to end the program")

print(" ")

m = mode\_select()

mode = 'w' **if** m=='write' **else** 'a'

filename = 'myfile.txt'

lineNumber = 0

final\_text = ''

**with** open(filename,mode) **as** file:

**while** **True** :

lineNumber += 1

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

userText = input(userPrompt)

**if** userText == "STOP" :

**break**

file.write(userText)

file.write("**\n**")

final\_text = final\_text + '**\n**' + userText

print(userText)

file.close()

print(f"Wrote to the file '*{filename}*': **\n\n**"+final\_text)

**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>