**Level 1: File Handling Definitions**

Use the following resources to answer the questions about file handling in Python.

* <https://www.pythonforbeginners.com/files/reading-and-writing-files-in-python>
* <https://www.pythonforbeginners.com/cheatsheet/python-file-handling>

1. Explain the function of each of the following file handling commands
   1. The open() function

The open() function opens another file for use in Python. The syntax to open it is file\_object=open(“filename”,”mode”)

* 1. The read() method

The read() method is used to read a text file in Python. The full code for python to read is:

file.open(“testfile.txt”,”r”)

print file.read()

* 1. The readline() method

The readline() method reads one line of a file. There is also a readlines() method, which reads multiple lines at a time. The full code for the readline() method is:

file=open(“testfile.txt”, “r”)

print file.readline[s](3)

The “s” is only needed for multiple lines, so that’s why I put it in brackets.

* 1. The write() method

The write() method is used to add information or content to an existing file. The program used is the following:

file.write(“This is a test”)

* 1. The close() method

The close() method is used to close a file when done changing or using the file. It closes the file completely, terminates resources in use, and any further attempts to use the file object will fail. The program used is the following:

fh.close()

1. Research and explain the “Mode” used to open files in a Python program.
   1. ‘r’ mode

‘r’ mode opens the file in read-only mode. It starts reading from the beginning of the file.

* 1. ‘w’ mode

‘w’ mode opens a file in write-only mode. The pointer is placed at the beginning of the file and this will overwrite any existing file with the same name. It will create a new file if one with the same name doesn’t exist.

* 1. ‘a’ mode

‘a’ mode opens a file for appending new information to it. The pointer is placed at the end of the file. A new file is created if one with the same name doesn’t exist.

* 1. ‘r+’ mode

‘r+’ mode opens a file for reading and writing. It places the pointer at the beginning of the file.

* 1. Explain when and where the mode is used in a Python program

The mode is used when a file is opened.

<https://stackabuse.com/file-handling-in-python/>

1. Provide example code which opens a text file for reading and prints the contents of the file to the console display.
   1. Explain what each line of the program does.

f=open(“myfile.txt”,”r”)

print(f.read())

The first line defines the file. The second file prints the file text.

1. Provide example code which opens a text file for writing and writes some data to the file.
   1. Explain what each line of the program does.

f=open(“myfile.txt”,”a”)

f.write(“Hows up?”)

f.close()

The first line opens a file. The second line writes the text. The third line closes the file.

1. Research and explain the difference between a “File Name” (type Python string) and   
   a File Object (type Python object).

Python Strings can be output to screen using the print function. A File Object is an object-oriented programming language. Almost everything in Python is an object, with its properties and methods.

**Level 2: Reading & Writing Files**

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. A sample file contents could look like:

*Hello kind student*

*This is a message from your computer*

*I hope you are having fun learning to program*

*Remember to ask Mr. Nestor questions when you don’t understand*

My text:

Hello Louise Arbour Secondary School

I am here to tell you all some great news

School is cancelled for the rest of the semester

I hope you all enjoy

1. Write a program that opens “myfile.txt” for reading and prints the contents to the file to the console display.
   1. The program should also print out the number of lines in the file
   2. Provide a listing of your program below

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

print(f.read())

def line\_count(myfile):

infile = open(myfile).readlines()

return len(infile)

print("Total Number of Lines:")

print(line\_count('myfile.txt'))

1. Write a program that opens “myfile.txt” for appending new contents to the file.
   1. You can “hard code” some commands to write new text to the file
   2. Make sure to use the close() method when your are finished.   
      (What happens if you don’t?)

If you don’t use the close() method, then it doesn’t do anything to the file.

* 1. How can you tell that your program worked? (That the contents changed?)

You can go onto the file to check if the context has changed.

* 1. Provide a listing of your program below

f=open("myfile.txt","a")

f.write(" My name is Harman")

f.close()

1. Write a program that opens “myfile.txt” for writing new contents to the file.
   1. You can “hard code” some commands to write new text to the file
   2. Explain the difference between appending and writing to a file.

Appending adds lines while writing deletes all data and adds new data.

* 1. Provide a listing of your program below

f=open("myfile.txt","w")

f.write("Hi World")

f.write("I hope you like my Hacking")

f.close()

**Level 3: Folders & Binary Files**

1. Add a folder called “resources” to your project as follows:
   * Click on “Add Folder” icon in the files pane/window.
   * Type “resources” and return.
2. Drag and drop your “myfile.txt” file into the “resources” folder.
3. Run you program from Level 2 to see what happens.
   1. Why does it give an error?

It gives an error as the file is now in a folder rather than just an individual.

* 1. How can you modify the file name string used by the open() function so that it also includes the “resources” folder?

You add the folder name before the name of the file. So it would be for example:

F=open(“resources/myfile.txt”,”r”).

* 1. Fix the open() function so that the program runs correctly and provide your program listing below.

f=open("resources/myfile.txt","r")

print(f.read())

f.close()

1. Research and explain the “Binary Mode” used to open files in a Python program.
   1. What is the ‘rb’ mode and how is it different from the ‘r’ mode

‘rb’ mode opens the file as read-only mode in binary format and starts reading from the beginning of the file. It is different from the ‘r’ mode as it stores data in forms of bits(0 or 1) rather than text files data.

* 1. What is the ‘wb’ mode and how is it different from the ‘w’ mode

‘wb’ mode opens a write-only file in binary mode. It is just as different from ‘w’ mode as ‘rb’ mode is from ‘r’ mode.

1. Add the “Penguin.bmp” binary image file to your repl project as follows:
   1. Download the “Penguin.bmp” file from the GitHub repository to your desktop
   2. Drag and drop the “Penguin.bmp” from your desktop to the “resources” folder in your repl project
   3. Click on the “Penguin.bmp” to make sure everything is ok.
2. Modify your Level 2 program to open the “Penguin.bmp” and print its contents to the screen.
   1. Provide a listing of your modified code below

f=open("resources/Penguin.bmp","rb")

print(f.read())

f.close()

* 1. Explain what you see as output compared to the penguin image itself

It gives up with numbers and letters rather than the image. For example, the last line says:

xff\xc0\x00\x00\x00’