**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

Used to open files in the system

* 1. The read() method

Returns one big string

* 1. The readline() method

Returns one line at a time

* 1. The write() method

Used to write a certain set of characters into a file

* 1. The close() method

Closes a file and clears up space it was taking up

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

Read mode is used when the file is only being read

* 1. ‘w’ mode

Write mode allows you to add info to the file

* 1. ‘a’ mode

Appending mode which allows you to add new data to the end of the file

* 1. ‘r+’ mode

Special read a write mode that handles both actions when working on a file

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

r- to have a file open for further code

w- adding to file

a-adding info to the end of the file

r+-used to read and write info to the document

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(“workfile", “r”)**----------------this line will open the file and let the computer know what we want to do with it

**Print F**----------------------this line will print the file contents and put it on the console screen

**F.close()**----------------------this line will close the file

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.

**File=open(“workfile”, “w”)** ------------------------------------------- this line will open the file and let the computer know what we want to do with it

**File.write(“Hello World”)**--------------------------------this line will add the text “Hello World” to the file

**File.close()()**----------------------this line will close the file

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

A Filename will call the file, and is a name for the file, whereas the File Object is a return once you’ve called the Filename which then gets used with the mode that you’ve assigned to the File Object.

**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\n*

*This is a message from your computer\n*

*I hope you are having fun learning to program\n*

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

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

1.File=open("myfile.txt", "r")

2.print(File.read())

1. File.close
2. 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?)

Then the next action that you assign will make changes to the file selected beforehand. When you close the file, nothing else can be changed to it.

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

By looking at the file and seeing the new text

* 1. Provide a listing of your program below

1. File=open("myfile.txt", "a")
2. File.write("This is a new line in the file")
3. File.close()
4. 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 will add the text to the end of the file but writing will replace all the writing with the new ones

* 1. Provide a listing of your program below

1. File=open("myfile.txt", "w")
2. File.write("This is a new line in the file")
3. File.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’