**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 returns a file object so it’s used with an argument. An argument is nothing more than a value that has been provided to a function, which is relayed when you call it.

* 1. The read() method

 extracts a string that contains all characters in the file

* 1. The readline() method

Reading a file line by line

* 1. The write() method

Write a sequence of strings to the file.

* 1. The close() method

When done with a file use “close()” to close it and free up the system.

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

Read mode, use when you want to read the file.

* 1. ‘w’ mode

Write mode, used to rewrite and edit the file.

* 1. ‘a’ mode

Appending mode, which is used to add new data to the end of the file; that is new information is automatically amended to the end

* 1. ‘r+’ mode

Special read and write mode, which is used to handle both actions when working with a file

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

‘r’ mode is used when reading a file, ‘w’ mode is used when editing files, ‘a’ mode is used when adding new data at the end and ‘r+’ mode can read and write.

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.

file = open("test.txt","r")

print(file.read())

file.close()

The first line opens the text file in read mode. The second line prints the text in the file to the console. The third line closes 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("test.txt","w")

file.write("Harkirat was here")

file.close()

The first line opens the file in write mode. The second line changes the text in the file to Harkirat was here. The last line closes the file.

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

A file objects are gained when you use the open function. It allows for alteration of the file name.

**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*

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

file = open("test.txt","r")

for x in file:

print(x)

file.close()

My answer:

fH = open("file.txt","r")

numLines = 0

for line in fH :

print(line)

numLines += 1

print("Number of lines is ", numLines)

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

* 1. Make sure to use the close() method when your are finished.   
     (What happens if you don’t?)

The new text is not written to the file.

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

You can check if the program worked by checking the contents of the file.

* 1. Provide a listing of your program below

file = open("test.txt","a")

file.write("New text")

file.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.

Writing to a file replaces any previous information while appending adds new content.

* 1. Provide a listing of your program below

file = open("test.txt","w")

file.write("New text")

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?

The file is no longer in the location that it previously was

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

You can modify the string so that it includes the open function by adding it before the file name when opening a file.

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

file = open("resources/test.txt","r")

print(file.read())

file.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 is used to read binary files which is anything that is not a text file.

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

‘wb’ mode is used to write on binary files.

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

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

print(file.read())

file.close()

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

I see a large amount of binary text. It consists of characters seperated by ‘/’s.