**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
   2. The read() method
   3. The readline() method
   4. The write() method
   5. The close() method
2. Research and explain the “Mode” used to open files in a Python program.
   1. ‘r’ mode
   2. ‘w’ mode
   3. ‘a’ mode
   4. ‘r+’ mode
   5. Explain when and where the mode is used in a Python program
3. 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.
4. 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.
5. Research and explain the difference between a “File Name” (type Python string) and   
   a File Object (type Python 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
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?)
   3. How can you tell that your program worked? (That the contents changed?)
   4. Provide a listing of your program below
3. 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.
   3. Provide a listing of your program below

**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?
   2. How can you modify the file name string used by the open() function so that it also includes the “resources” folder?
   3. Fix the open() function so that the program runs correctly and provide your program listing below.
4. 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
   2. What is the ‘wb’ mode and how is it different from the ‘w’ mode
5. 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.
6. 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
   2. Explain what you see as output compared to the penguin image itself