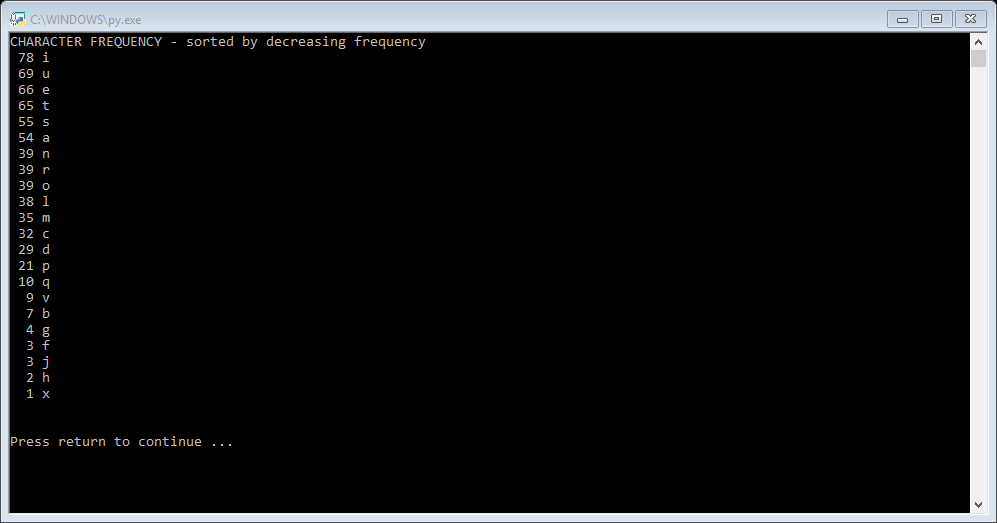
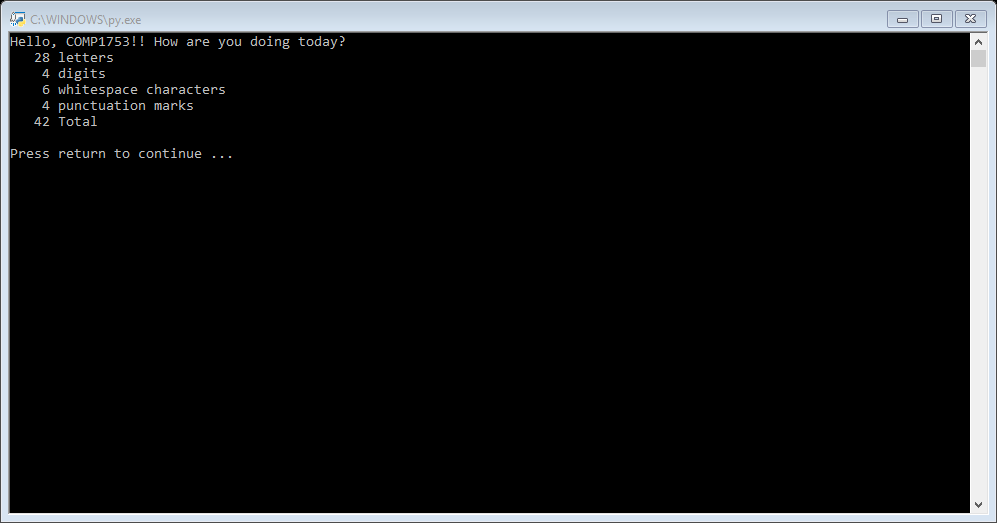
# Tutorial 08 – Strings

This tutorial is aimed at familiarising you with using strings. It will also test you on the use of string-based functions, methods and loops.

Do not worry if you can’t do all the exercises, especially the difficult ones. Give them a try and if you are stuck, ask your tutor.

For each exercise you should make a copy of the example you are working on **before you modify it**. In that way you will have both your new version and the original version to compare it with.

You should make a record all of your work in your COMP1753 logbook.

1. Download the examples and last week’s solutions. Check through the solutions and make sure you could do them all**.** Unzip this week’s examples.
2. Make a copy of 14String\_characterFrequency and change the name to 14Lorem\_characterFrequency. Now modify it so that it counts the character frequency, sorted in descending order, for the “Lorem ipsum” text, rather than the “Hello world” string. The output will not be neatly lined up, so modify the code using the **format()** function to right-justify all the numbers. Your output should look something like this:
3. Write a program which loops repeatedly and asks the user for a string each time. The program should do an analysis, similar to that in 13Lorem\_counts, to count the number of characters, letters and words in the string. When the user enters an empty string, the loop should terminate. [Hint: use while True: to run the loop indefinitely and break to terminate it.]
4. The **remove\_whitespace()** and **remove\_punctation()** functions can be written as a single function using the **.isalnum()** method to eliminate anything that isn’t alpha-numeric (this is more efficient, as it uses a single function, but less flexible because, for example, it can’t remove just the punctuation and leave the whitespace). Write a program similar to 12String\_removeFunctions which removes all the whitespace and punctuation in single function. Test it with the “Hello world!! How are you doing today?” string. [Hint: you will need to loop over the string, rather than the punctuation / whitespace strings.]
5. Now build a program with 4 separate functions which count the number of letters, digits, whitespace characters and punctuation marks, respectively. Test your functions with the string “Hello COMP1753!! How are you doing today?”, using the **format()** function to right-justify all the numbers. The output should look something like this:
6. Read the w3schools page mentioned in the lecture :
   1. <https://www.w3schools.com/python/python_dictionaries.asp>