I/O and File Handling

Exercises

Week 8

Prior to attempting these exercises ensure you have read the lecture notes and/or viewed the video, and followed the practical. You may wish to use the Python interpreter in interactive mode to help work out the solutions to some of the questions.

Download and store this document within your own filespace, so the contents can be edited. You will be able to refer to it during the test in Week 6.

Enter your answers directly into the highlighted boxes.

For more information about the module delivery, assessment and feedback please refer to the module within the MyBeckett portal.

Which of the following represents a Python *f-string*?

```
a) "Hello {}, you have logged in".format(name)

b) "Hello {name}, you have logged in"

c) f"Hello {name}, you have logged in"

d) "Hello %s, you have logged in" % name
```

Answer:

c) f"Hello {name}, you have logged in"

_____

Given the following definition of `value`, what would each of the following statements display?

```
value = 10.768572

print(f"Value is {value}")
```

Answer:

value is 10.768572

```
print(f"Value is {value * 10}")
```

Answer:

value is 107.68572

```
print(f"Value is {value:.2f}")
```

Answer:

value is 10.77

```
print(f"Value is {value:16.2f}")
```

Answer:

value is 10.77

```
print(f"Value is {value:0>16.2f}")
```

Answer:

> value is 000000000010.77

Within an *f-string* **format specifier** what does the '^' alignment character signify?

Answer:

> Center-aligns the text.

Write a statement which uses the `str.format()` to generate the same output as the following *f-string* statement -

```
print(f"pi to 5 decimal places is {math.pi:.5f}")
```

Answer:

> print("pi to 5 decimal places is {:.5f}".format(math.pi))

What would the following statement display?

```
print("Length = {1} Width = {0}".format(10,20))
```

Answer:

> Length=20
> Width=10

What *exactly* would the following statement display?

```
print("Hello".rjust(10))
```

Answer:

> Its ' Hello'

On which older programming language is the *%-formatting* style loosely based?

Answer:

Its based on C.

Write a Python program that uses a loop and the `str.rjust()` method to generate the following output.

```
##########
 #########
  ########
   #######
    ######
     #####
      ####
       ###
        ##
         #
```

Hint: The program will start as follows

```
for n in range(10,0,-1):
    line = "#" * n
    # rest of code....
```
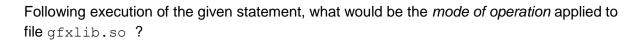
Answer:

```
for n in range(10, 0, -1):
    line = "#" * n
    print(line.rjust(10))
```

What is the basic element that *all* computer files contain?

Answer:

The basic elements is Byets.

What *function* must be called before the contents of a file can be accessed?

Answer:

```
open()
```

What *method* must be called on a file object once processing is complete?

Answer:

```
close()
```

Following execution of the given statement, would the file 'myfile.txt' be open for *reading* or for *writing*?

```
f = open("myfile.txt")
```

Answer:

Reading (default mode is "r")

Following execution of the given statement, would the file yourfile.txt be open for *reading* or for *writing*?

```
f2 = open("yourfile.txt", "w")
```

Answer:

writing

Following execution of the given statement, what would be the *mode of operation* applied to file `gfxlib.so` ?

```
f3 = open("gfxlib.so", "r+b")
```

Answer:

Reading and writing both in binary mode.

_____

What is the difference between the two following method calls?

```
f.readline()
f.readlines()
```

Answer:

f.readline() it reads a single line from the file.
f.readlines() reads all lines and returns them as a list of string.

_____

How much of the file content would be read with the following method call?

```
content = f.read()
```

Answer:

The entire file contain.

_____

If the variable 'my_file' referred to a text file, what would the following code do?

```
for next in my_file:
    print(next)
```

Answer:

It reads and prints each line of the file one by one, including newline characters.

_____

What is the issue with the following code? And how could it be fixed?

```
f = open("details.txt", "w")
total = 100
f.write(total)
f.close()
```

Answer:

The issue is total is an integer, but f.write() expects a string.

Fixing: we should write it:
f.write(str(total))

_____

What is the purpose of the file `tell()` method?

Answer:

It returns the current position of a cursor.

_____

What does the following code do?

```
f.seek(0)
```

Answer:

It resets the place where the file print will start in the given file, to the start of the file.

_____

Why is file handling often done using a `with` statement as shown below?

```
with open("data.txt") as f:
     lines = f.readlines()
```

Answer:

It makes sure that the file will always be closed upon performing its operations, even when an error is encountered.

Exercises are complete

Save this logbook with your answers. Then ask your tutor to check your responses to each question.