Exercise 4 – String handling

Objective

To consolidate string manipulation in Python. This includes further practise at general Python constructs, such as loops.

Questions

1. Open the script **sep.py** in a text editor. You'll see a string defined called 'Belgium'. Add code to print:
    1. A line of hyphens the same length as the Belgium string, followed by...
    2. the string with the comma separators replaced by colons ':'. Followed by...
    3. the population of Belgium (the second field), **plus** the population of the capital city (the forth field).

        Hint: The answer should be 11183818.
    4. Add a line of hyphens the same length as the Belgium string.

If time allows...

2. Examine the file **messier.txt** in the **labs** directory, which contains details of celestial 'Messier' objects. It consists of several columns for each object, identified by the 'M' number. The columns are as follows:

    MessierNumber CommonName ObjectType Constellation

    Note that many have no common name. Read the file using a **for** loop:

    ```
    for line in open('messier.txt', encoding='latin_1'):
     if not line: break
       # The text is in the variable named 'line'
    ```

Ignore lines that do not start with 'M'. Print the fields from each line delimited with '|' characters. Where there is no common name, use 'no name'. Ignore any lines not beginning with a Messier number. For example:

|M1|The Crab Nebula|Supernova remnant|Taurus|

|M2|no name|Globular cluster|Aquarius|

|M3|no name|Globular cluster|Canes Venatici|

Hint: The header on the file should assist in getting the field positions.

Solutions

Question 1

a) A line of hyphens the same length as the Belgium string, followed by...

b) the string with the comma separators replaced by colons ':'. Followed by...

c) the population of Belgium (the second field), **plus** the population of the capital city (the fourth field). **Hint:** The answer should be 11183818.

If you did this:

```
print(items[1] + items[3])
```

then you would've got string concatenation, and an apparently very large number! You need to change each value to an int.

d) Add a line of hyphens the same length as the Belgium string.

```
items = Belgium.split(',')
print('-' * len(Belgium))                  # a)
print(':' . join(items))                   # b)
print(int(items[1]) + int(items[3]))       # c)
print('-' * len(Belgium))                  # d)
```

If time allows...

Question 2

```
for line in open('messier.txt'):
    if not line: break
    if line.startswith('M'):
        # Slice each field
        mes_num = line[:6].rstrip()
        com_name = line[6:40].rstrip()
        if not com_name: com_name = 'no name'
        obj_type = line[40:64].rstrip()
        const = line[64:].rstrip()
    print(f"|{mes_num}|{com_name}|{obj_type}|{const}|")
```