The *format* method

Suppose that you have three lists: one containing integers, one containing strings, and one containing floating point numbers. Suppose further that you want to print these lined up in columns, as in this example:

```
    5      Sofia   12.304
  101   Isabella    3.000
   40     Camila  698.039
   33  Valentina    4.900
   12    Valeria  -45.831
  101    Mariana   10.000
    4   Gabriela   -4.040
```

That is, you want to print them right-justified in three columns.

To do so, you would use the ***format*** method. It works like this example:

```
'blah blah {:6d} xxx {:>8}yyy {:6.2f}'.format(40, "hello", 4.3)
```

The thing before the DOT is a string that we call the FORMATTING string. The ***format*** method returns a string that is the same as the formatting string, but with the things in ***curly-braces {..}*** replaced by the ***arguments*** to the ***format*** method. Furthermore, those arguments are formatted per the specification inside the curly-braces.

So in the above example, the returned value is:

```
'blah blah     40 xxx    helloyyy   4.30'
```

Note that:

- The non-curly-brace part of the formatting string is returned unchanged.
- The integer `40` was placed in a field of 6 spaces because of the `{:6d}`.
- The string `"hello"` was placed in a field of 8 spaces, right-justified, because of the `{:>8}`.
- The floating point number `4.3` was placed in a field of 6 spaces, with two digits after the decimal point, because of the `{:6.2f}`.

The details of the formatting are not important here (you can look them up). All you need to understand is that the ***format*** method returns its formatting string with the curly-braces inside the formatting string being replaced by the arguments to the ***format*** method:

```
FORMATTING_STRING.format(blah, blah, blah, ...)
```

Returning to the first example above, the following code would produce the table of three columns, assuming that the three lists each have the same length:

```
for k in range(len(list_of_ints)):
    print('{:4d} {:>10s} {:8.3f}'.format(list_of_ints[k],
                                         list_of_strings[k],
                                         list_of_floats[k]))
```

You will see the above explanation repeated along with details in your subsequent reading.