Python #5

We have continued learning about **string formatting**, seen several options to get a formatted string - positional, named, a combination of both. Also we have an **f-string** which is the shortest way to build a string with embedded data. It is possible to add formatting type inside placeholders to get the desired look of data.

If some repetitive work needs to be done, **loops** can come in handy. **For-loops** are specially designed to work with **collections** (like **lists**)and **strings**. For needs other than just sweeping the collection, like a better access control using index, **range** function can help and create the desired sequence. If we get more collection dimensions like list of lists, adding an inner **for-loop** can sweep them forming **nested for-loop**. Another means of flow control in Python are **break** and **continue** statements in **for-loop**.

More topics covered:

* **join** method

Links:

* [string formatting](https://www.w3schools.com/python/ref_string_format.asp)
* [f-strings](https://realpython.com/python-string-formatting/)
* [range function](https://www.w3schools.com/python/ref_func_range.asp)
* [for loops](https://www.w3schools.com/python/python_for_loops.asp)