List Comprehension

List Comprehension

There will be many times you will want to automate the creation of a list. You can use loops for this but can become impractical. A nice way to generate lists is using **list comprehension**:

```python
[2]: #Generating a list of integers in ascending order
     numbers = [i for i in range(6)]
     print(numbers)
```

```
[0, 1, 2, 3, 4, 5]
```

You can treat the `for` inside the list just like a `for` loop, including looping through collections:

```python
[4]: string = 'abcdefg'

     #Generating a list of characters from a string
     char_list = [char for char in string]
     print(char_list)
```

```
['a', 'b', 'c', 'd', 'e', 'f', 'g']
```

Only use list comprehension if you are interested in the list itself. Do not use it in place of a `for` loop.

You can also embed list comprehension:

```python
[5]: print([[i + j for j in range(3)] for i in range(4) ])
```

```
[[0, 1, 2], [1, 2, 3], [2, 3, 4], [3, 4, 5]]
```