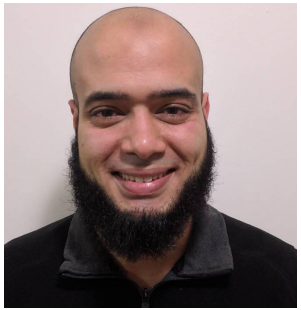
Python Programming

Nesting with Comprehension

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)



Flatten a list

```
2
3   lst_of_lists = [[1, 2], [3], [4, 5, 6, 7, 8], [9, 10, 11]]
4
5   # Flatten a list: make all the items in a single list with no inner list
6   # we can do that easily with list comprehension
7
8   # without comprehension
9   lst1 = []
10  for lst in lst_of_lists:
11      for item in lst:
12          lst1.append(item)
13
14  print(lst1)
15  # [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
```

```
lst2 = [item for lst in lst_of_lists for item in lst]
# same list!
```

Add int to all

```
3   # without
4   def add_value(lst_of_lists, value):
5       # add the value to each item
6       new_lst_of_lists = []
7       for lst in lst_of_lists:
8           new_lst = []
9           for item in lst:
10              new_lst.append(item + value)
11          new_lst_of_lists.append(new_lst)
12      return new_lst_of_lists
13
14  lst_of_lists = [[1, 2], [3], [4, 5, 6, 7, 8], [9, 10, 11]]
15
16  value = 10
17  print(add_value(lst_of_lists, value))
18  # [[11, 12], [13], [14, 15, 16, 17, 18], [19, 20, 21]]
```

```
# we get the lst, then transform it to a new list
lst_of_lists2 = [ [item+value for item in lst] for lst in lst_of_lists]
print(lst_of_lists2)
# [[11, 12], [13], [14, 15, 16, 17, 18], [19, 20, 21]]
```

Generating Pairs

```
# we know with zip we can create iterator over item from each
# what if we want all pairs

lst1 = [1, 2]
lst2 = [10, 20, 30]

# without comprehension
lst_pairs1 = []
for item1 in lst1:
    for item2 in lst2:
        lst_pairs1.append((item1, item2))
print(lst_pairs1)   # [(1, 10), (1, 20), (1, 30), (2, 10), (2, 20), (2, 30)]
```

```
lst_pairs2 = [(item1, item2) for item1 in lst1 for item2 in lst2 ]
```

Creating simple grids!

```
3   # How to create an 3x4 grid of some value (e.g. 0)?
4
5   rows, cols = 3, 4
6   lst = [[0] * rows] * cols
7   print(lst)
8   # [[0, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]]
9
10  lst[0][0] = 2
11  print(lst)
12  # [[2, 0, 0], [2, 0, 0], [2, 0, 0], [2, 0, 0]]  hmmm
13  print(id(lst[0]), id(lst[1]))   # 0x111 0x111
14  # * cols just append the same object
15
16  lst = [ [0] * rows for i in range(cols) ]
17  lst[0][0] = 2
18  print(lst)
19  # [[2, 0, 0], [0, 0, 0], [0, 0, 0], [0, 0, 0]]
20
21  lst = [[x for x in range(rows)] for y in range(cols)]
22  print(lst)
23  # [[0, 1, 2], [0, 1, 2], [0, 1, 2], [0, 1, 2]]
24
```

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."