

0098_list_comprehsions_map_filter

November 10, 2018

1 Applying functions to lists, and filtering lists with list comprehensions, map and filter

These examples are intended as a reminder of how to use list comprehensions, map and filter. They are not intended to be an exhaustive tutorial.

Let's start with a list of numbers, and we wish to:

- 1) Square all numbers
- 2) Find even numbers
- 3) Square all even numbers

That is where list comprehensions, map and filter come in. There is significant overlap between these methodologies, but here they are.

1.1 Define my list of numbers

```
In [1]: my_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

1.2 Square numbers with list comprehension

```
In [2]: answer = [x**2 for x in my_list]
        print (answer)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

1.3 Map a named lambda function to square the numbers

We use a one-line lambda function here, but a full function with define and return could also be used.

```
In [3]: sq = lambda x: x**2
        answer = list(map(sq, my_list))
        print (answer)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

1.4 Map a lambda function directly to square numbers

```
In [4]: answer = list(map(lambda x: x**2, my_list))
        print (answer)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

1.5 Use a lambda function in a list comprehension

```
In [5]: sq = lambda x: x**2
        answer = [sq(x) for x in my_list]
        print (answer)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

1.6 Filter a list with list comprehension to find even numbers

For even numbers $x\%2$ will equal zero:

```
In [6]: answer = [x for x in my_list if x%2 == 0]
        print (answer)
```

```
[2, 4, 6, 8, 10]
```

1.7 Filter a list with a named lambda function

Full functions call also be used. The function (full or lambda must return True or False)

```
In [7]: is_even = lambda x: x%2 == 0
        answer = list(filter(is_even, my_list))
        print (answer)
```

```
[2, 4, 6, 8, 10]
```

1.8 Filter a list with a lambda function applied directly

```
In [8]: answer = list(filter(lambda x: x%2 ==0, my_list))
        print (answer)
```

```
[2, 4, 6, 8, 10]
```

1.9 Combine squaring and filtering with a list comprehension

List comprehensions may filter and apply a function in one line. To get the square of all even numbers in our list:

```
In [9]: answer = [x**2 for x in my_list if x%2 == 0]
        print (answer)

[4, 16, 36, 64, 100]
```

Or we may apply lambda (or full) functions in both the operation and the filter in a list comprehension.

```
In [10]: sq = lambda x: x**2
         is_even = lambda x: x%2 == 0
         answer = [sq(x) for x in my_list if is_even(x)]
         print (answer)

[4, 16, 36, 64, 100]
```

1.10 Combine squaring and filtering with map and filter

filter and map would need to be used in two statements to achieve the same end. Here we will use named lambda functions, but they could be applied directly as above.

```
In [11]: sq = lambda x: x**2
         is_even = lambda x: x%2 == 0

         filtered = list(filter(is_even, my_list))
         answer = list(map(sq, filtered))
         print (answer)

[4, 16, 36, 64, 100]
```