Forward School

Program Code: J620-002-4:2020

Program Name: FRONT-END SOFTWARE DEVELOPMENT

Title : Exercise 3 - List Comprehension & Lambda

Name: Ooi Caaron

IC Number: 990701-07-5837 ¶

Date : 24/6/23

Introduction : Using List Comprehension and Lamda to return the output

Conclusion : Still need to do a lot practice

EXERCISE 3

List Comprehension & Lambda Exercise

In [4]:

```
# write list comprehension to determine the length of each word
# except 'the' and store as 'word_lengths'
sentence = "the quick brown fox jumps over the lazy dog"
word_lengths = [len(w) for w in sentence.split() if w != "the"]

word_lengths
```

Out[4]:

```
[5, 5, 3, 5, 4, 4, 3]
```

In [10]:

```
# write a list comprehension to extract the
# negative numbers from the list as integers and store as newlist
numbers = [34.6, -203.4, 44.9, -68.3, -12.2, 44.6, 12.7]
negative_nums = [round(n) for n in numbers if n < 0]

negative_nums
```

Out[10]:

```
[-203, -68, -12]
```

In [17]:

```
# Convert the following code to list comprehension

coords = [(x, y) for x in range(4) for y in range(2)]

for x in range(4):
    for y in range(2):
        coordinate = (x, y)
        coords.append(coordinate)
print(coords)
```

```
[(0, 0), (0, 1), (1, 0), (1, 1), (2, 0), (2, 1), (3, 0), (3, 1), (0, 0),
(0, 1), (1, 0), (1, 1), (2, 0), (2, 1), (3, 0), (3, 1)]
```

In [32]:

```
# write a list comprehension to list all the combinations
# for the two sets of words

set1 = ['ball','cheese','round']
set2 = ['cake','rice','ham']

combinations = [(word1, word2) for word1 in set1 for word2 in set2]
print(combinations)
```

```
[('ball', 'cake'), ('ball', 'rice'), ('ball', 'ham'), ('cheese', 'cake'),
('cheese', 'rice'), ('cheese', 'ham'), ('round', 'cake'), ('round', 'ric
e'), ('round', 'ham')]
```

In [31]:

```
# write a lambda function that squares the number
# for all odd numbers from 1 to 100
x = range(1,101)
odd_nums = [n*n for n in filter(lambda n: n % 2 != 0, x)]
print(list(odd_nums))
```

```
[1, 9, 25, 49, 81, 121, 169, 225, 289, 361, 441, 529, 625, 729, 841, 961,
1089, 1225, 1369, 1521, 1681, 1849, 2025, 2209, 2401, 2601, 2809, 3025, 32
49, 3481, 3721, 3969, 4225, 4489, 4761, 5041, 5329, 5625, 5929, 6241, 656
1, 6889, 7225, 7569, 7921, 8281, 8649, 9025, 9409, 9801]
```

In [30]:

```
# write a list comprehension that squares number
# for all odd numbers from 1 to 100
x = range(1,101)
odd_nums = [n*n for n in x if n % 2 != 0]
print(odd_nums)
```

```
[1, 9, 25, 49, 81, 121, 169, 225, 289, 361, 441, 529, 625, 729, 841, 961,
1089, 1225, 1369, 1521, 1681, 1849, 2025, 2209, 2401, 2601, 2809, 3025, 32
49, 3481, 3721, 3969, 4225, 4489, 4761, 5041, 5329, 5625, 5929, 6241, 656
1, 6889, 7225, 7569, 7921, 8281, 8649, 9025, 9409, 9801]
```

In [37]:

```
# write a lambda function to extract names that begin with 'A'
names = ['Anne', 'Amy', 'Bob', 'David', 'Carrie', 'Barbara', 'Zach']
names_b = filter(lambda n : n[0] == "B",names)
print(list(names_b))
```

```
['Bob', 'Barbara']
```

In [34]:

```
# write a list comprehension to extract names that begin with 'B'
names = ['Anne', 'Amy', 'Bob', 'David', 'Carrie', 'Barbara', 'Zach']
names_b = [n for n in names if n[0] == "B"]
print(names_b)
```

```
['Bob', 'Barbara']
```

In [ ]: