الاسم : احمد السيد عبده على فرج | سكشن 1

Assignment 1

Transform and Clean Data

You are given a list of product names that contain extra spaces and inconsistent capitalization:

products = [" LAPTOP ", "phone ", " Tablet", "CAMERA "]

Using `map()` and `lambda`, clean the data by:

Removing extra spaces

Converting to title case (e.g. `"laptop"` → `"Laptop"`)

Expected Output:

['Laptop', 'Phone', 'Tablet', 'Camera']

Answer

```
products = [" LAPTOP ", "phone ", " Tablet", "CAMERA "]
print(list(map( lambda p : p.strip().title(), products)))
```

Convert Temperatures (Celsius → Fahrenheit)

Given a list of temperatures in Celsius, convert them to Fahrenheit using:

F=9/5C+32

celsius = [0, 10, 20, 30, 40]

Expected Output:

[32.0, 50.0, 68.0, 86.0, 104.0]

Hint:

Use map(lambda c: (9/5)*c + 32, celsius)

Answer

```
celsius = [0, 10, 20, 30, 40]
print(list(map(lambda c : (9 / 5) * c + 32 , celsius
```

Apply Multiple Transformations

You have a list of integers. You need to:

Square each number.

Then add 10 to each result.

All using **one `map()` and one `lambda`**.

nums = [1, 2, 3, 4, 5]

Expected Output:

[11, 14, 19, 26, 35]

Hint:

Combine both operations in one lambda.

Answer

```
nums = [1, 2, 3, 4, 5]
print(list(map(lambda n : n ** 2 + 10, nums)))
```

Extract First and Last Characters

Given a list of words, create a new list of tuples `(first_char, last_char)` for each word.

words = ["python", "lambda", "programming", "map", "function"]

Expected Output:

[('p', 'n'), ('l', 'a'), ('p', 'g'), ('m', 'p'), ('f', 'n')]

Hint:

Use string indexing in the lambda: `lambda w: (w[0], w[-1])`

Answer

```
words = ["python", "lambda", "programming", "map",
"function"]
print(list(map(lambda w : (w[0] , w[-1]) , words)))
```

marks = [[45, 80, 70], [90, 60, 100], [88, 76, 92]]

Using nested `map()` and `lambda`,

Increase each mark by 5%,

Round it to the nearest integer.

Expected Output:

[[47, 84, 74], [95, 63, 105], [92, 80, 97]]

Hint:

Use:

map(lambda row: list(map(lambda x: round(x * 1.05), row)), marks)

Answer

```
marks = [[47, 84, 74], [95, 63, 105], [92, 80, 97]]

print(list(map(lambda row: list(map(lambda x: round(x *
1.05), row)), marks)
```

6- program that normalizes a list of numbers between 0 and 1 using `map()` and `lambda`.

Answer

```
numbers = [5, 10, 15, 20, 25]

min_val = min(numbers)
max_val = max(numbers)

normalized = list(map(lambda x: (x - min_val) / (max_val -
min_val), numbers))

print(normalized)
```

Given a list of sentences, extract the length of each word in every sentence using nested `map()`.

Answer

```
sentences =
 ["I love python" , "coding is fun" , "painting is cool"]

res = list(
    map(
        lambda sentence: list(
            map(lambda word: len(word), sentence.split())
        ),
        sentences
    )
)
Print(res)
```