

235229143_Swetha.K_labsheet_6 Programming and Data Structures with Python Lab6.

Implementation of Map, Filter and Reduce Functions

Question1

Write a program to implement MAP function. Find the square root of a list of numbers [1, 2, 4, 6] using map and sqrt functions. Check the answer against your user defined function mymap().

```
In [1]: def square_num(n):  
        return n * n  
        nums = [4, 5, 2, 9]  
        print("Original List: ",nums)  
        result = map(square_num, nums)  
        print("Square the elements of the said list using map():")  
        print(list(result))
```

Original List: [4, 5, 2, 9]

Square the elements of the said list using map():

[16, 25, 4, 81]

Question2

Write a program to implement FILTER function. Filter all upper case letters in a list ['x', 'Y', '2', '3', 'Z', 'b'] using filter function. Check the answer against your user define function myfilter().

```
In [2]: test_list = ['x', 'Y', '2', '3', 'Z', 'b']

# printing original list
print("The original list is : " + str(test_list))

res_list = []
for sub in test_list:
    res = True
    for ele in sub:

        # checking for uppercase
        if not ele.isupper():
            res = False
            break
    if res:
        res_list.append(sub)

print("Filtered Tuples : " + str(res_list))
```

The original list is : ['x', 'Y', '2', '3', 'Z', 'b']
 Filtered Tuples : ['Y', 'Z']

Question3

Write a program to create a lambda function that takes two characters and concatenates them. Now, apply this function inside REDUCE function that will reduce the list of characters ['a', 'b', 'c', 'd'] with the initial value 'x'.

```
In [8]: from functools import reduce
concatenate_chars = lambda x, y: x + y
initial_value = 'x'
character_list = ['a', 'b', 'c', 'd']
result = reduce(concatenate_chars, character_list, initial_value)
print(result)
```

xabcd

Question4

Imagine an accounting routine used in a book shop. It works on a list with sublists, which look like this: Order No Book Title and Author Quantity Price per Item 34587 Learning Python, Mark Lutz 4 40.95 98762 Programming Python, Mark Lutz 5 56.80 77226 Head First Python, Paul Barry 3 32.95 88112 Einführung in Python3, Bernd Klein 3 24.99

Write a Python program, which returns a list with 2-tuples. Each tuple consists of an order number and the product of the price per items and the quantity. The product should be decreased by RS 10 if the value of the order is smaller than RS 100.00. Write a Python program using lambda and map functions.

```

In [10]: order = [[34587, 'Learning Python', 'Mark Lutz', 4, 40.95],
                  [98762, 'Programming Python', 'Mark Lutz', 5, 56.80],
                  [77226, 'Head First Python', 'Paul Barry', 3, 32.95],
                  [88112, 'Einführung in Python3', 'Bernd Klein', 3, 24.99]
                  ]

#without using lambda, map, list comprehension

lists = []
for item in order:
    if item[-1]*item[-2] < 100:
        lists.append((item[0], item[-1]*item[-2]+10))
    else:
        lists.append((item[0], item[-1]*item[-2]))

print("Order Summary: ", lists)

#with using lambda and map
print("Order Summary: ", list(map(lambda x: (x[0], x[-1]) if x[-1]*x[-2] > 100 else

#with using list comprehension
print("Order Summary: ", [(item[0], item[-1]) if item[-1]*item[-2] > 100 else (item

Order Summary: [(34587, 163.8), (98762, 284.0), (77226, 108.85000000000001),
(88112, 84.97)]
Order Summary: [(34587, 40.95), (98762, 56.8), 108.85000000000001, 84.97]
Order Summary: [(34587, 40.95), (98762, 56.8), (77226, 108.85000000000001), (8
8112, 84.97)]

```

In []: