**Saroj Banjara**

**TASK FOUR: HIGHER ORDER FUNCTIONS, GENERATORS, LIST COMPREHENSION AND DECORATOR**

1. Write a program in Python to find the values which is not divisible 3 but it should be a multiple of 7. Make sure to use only higher order function.

x = list(filter(lambda x: not x % 3 == 0 and x % 7 == 0, range(1,100)))

print('x =', x) *#prints x = [7, 14, 28, 35, 49, 56, 70, 77, 91, 98]*

2. Write a program in Python to multiply the element of list by itself using traditional function and pass the function to map to complete the operation.

def multiply():

y = lambda x: x \* x

return y

result = map(multiply(), range(1, 10))

print(list(result))

*# prints [1, 4, 9, 16, 25, 36, 49, 64, 81]*

3. Write a program in Python to find out the character in a string which is uppercase using list comprehension.

string = 'aappleAPPleU'

x = [i for i in string if i.isupper()]

print('Uppercase letters :', list(x)) *#prints Uppercase letters : ['A', 'P', 'P', 'U']*

4. Write a program to construct a dictionary from the two lists containing the names of students and their corresponding subjects. The dictionary should maps the students with their respective subjects. Let’s see how to do this using for loops and dictionary comprehension. HINT-Use Zip function also

* Student = ['Smit', 'Jaya', 'Rayyan']
* capital = ['CSE', 'Networking', 'Operating System']

Using for loop:

student = ['smit','jaya','rayyan']

capital = ['CSE', 'Networking', 'OS']

result = {}

for key in student:

for value in capital:

result[key] = value

capital.remove(value)

break

print('result = ', result)

Using Dictionary Comprehension:

student = ['smit','jaya','rayyan']

capital = ['CSE', 'Networking', 'OS']

result = {student[i]:capital[i] for i in range(len(student))}

print(result)

Using zip()

student = ['smit','jaya','rayyan']

capital = ['CSE', 'Networking', 'OS']

result = dict(zip(student,capital))

print('result = ', result)

# returns {'smit': 'CSE', 'jaya': 'Networking', 'rayyan': 'OS'}

5. Learn More about Yield, next and Generators

A generator is a like a function, but whenever it needs to generate the value, it does so with the yield keyword rather than return keyword. And once a function has yield keyword on it, it is no more called as a function but as a generator. Generator function returns the generator object, hence in order to return a value, we use \_\_next\_\_() method.

6. Write a program in Python using generators to reverse the string. Input String = “Consultadd Training”

def gen():

string = 'Consultadd Training'

str1 = string[::-1]

yield str1

gen()

result = gen()

print(result.\_\_next\_\_()) *# prints ‘gniniarT ddatlusnoC’*

7. Write any example on decorators.

def check(func):

def inner(a, b):

if a < b:

a, b = b, a

if b == 0:

print("can't divide by zero!")

return

return func(a, b)

return inner

@check # we can also write div = check(div) underneath the div function.

def div(a, b):

return a / b

print(div(2, 10)) #returns 5.0

8. **Learn about What is FRONT END and its Technologies and Tools**

* Make sure to mention at least 5 top notch technologies of Frontend.
* Also mentioned the name of companies using those 5 technologies individually

What we see and interact with, as the visitors of a website, or a user of a mobile app is considered front-end technology. The front end is made up of many different languages and libraries. Top technologies and tools used in front-end design are HTML5/CSS3, Javascript, GIT, jQuery and Angular. Some of the websites which used above technologies to build the front-end are:

Dell - <https://www.dellemc.com/en-us/index.htm>

LS Productions - <https://www.dellemc.com/en-us/index.htm>