**Full Stack Data Science**

**Mukesh sir**

**Test 1**

Question 1:

s = [PythonProgramming]

if we see start =2 end =10 and step is 2

start 2: “t”

next : “o” step 2

next : "P’ step 2

next : "o” step 2

An answer might not be given in the options

Question 2:

Option a) heo

Answer should be heo they said to replace “l” with ‘’

Question 3:

Option b) [1,2,3,4,5]

Extend is used to extend the list provided to the back of the other list

Question 4:

Option: C tuple supports item assignment

Question 5:

Option : a

It gives tuple as op

Question 6:

Option : c) [30, 40, 50]

Last 3 elements

Question 7 :

Option : b

Reverse list

Question 8:

Option : a) [0, 4, 16]

Return square on even condition

Question 9:

Option : b) [x for x in range(10) x % 2 == 0]

Condition is not mentioned in list comprehension

Question 10:

OPTION a) [1, 3]

Starts with 1 ends by 5 step 2 gives this result

Question 11:

Option: d) [5]

starts with 5 prints that’s it

question 12 :

option: a

adds y=3 and z=4 to dict

question 13 :

option :a) {2: 8, 3: 27, 4: 64}

prints num and its cube

question 14:

option: b) “a”

returns 0th element which is a

question 15:

option: a) [1] [2]

returns a) [1] [2] elemnts

question 16:

option: b) 9

returns 9 by 3 \*3

question 17:

option: c) filter()

filter function is used to filter iterable

question 18:

option: b) 24

reduce function returns 24

question 19:

option: a) [3, 4]

condition is applied

question 20:

def fn(&a):

return a

1. Reverse a string without using slicing

def reverse\_string(s):

res=’’

for c in s:

c=c+res

return res

2. Write a function to remove duplicates from a list

sol

def remove\_duplicates(lst):

for num in lst:

if num not in lst

print(lst)

return(lst)

3. Write a dictionary comprehension that reverses keys and values

sol

def reverse\_dict(d):

{val, key in d}

4 Implement map() to find the cube of a list of numbers

Sol

Def cube(x):

Return x\*\*3

Lst=[1,2,3,4,5,6]

X=List(map(cube, lst)

Print(x)

5. Implement a function using filter() to remove vowels from a string

sol

def remove\_vowels(s):  
 vowel=”aeiou”

if s not in vowel:

return s:

s=” hello python”

x= list(remove\_vowels, s)

print(x)

6. Write a function that returns a dictionary of squares from 1 to n

Sol:

Def sq(num):

return num, num\*\*2

itr=[1,2,3,4]

x=dict(map(sq, itr)

print(x)

7. Write a function to merge two dictionaries

def merge\_dicts(d1, d2):

merge=d1.copy

merge.update(d2)

return merge

8. Implement reduce() to compute factorial of a number

from functools import reduce

def factorial(n,o):

# Your code here

Return n\*o

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

X= reduce( factorial, lis)

9. Write a function to flatten a nested list

def flatten\_list(nested\_lst):

flat=[]

for I in nested\_lst:

if isinstance(I, nested\_lst):

flat.extend (flatten\_lst(i))

else:

flat.append(i)

return flat

10. Write a lambda function to check if a number is prime

is\_prime = lambda n: n%i!=0 for i in range(2,n)