

Assignment Part-2

Q26. What is a string? How can we declare string in Python?

A string is a sequence of characters. A character is simply a symbol. For example, the English language has 26 characters

Strings in python are surrounded by either single quotation marks, or double quotation marks.

'hello' is the same as "hello"

Q27. How can we access the string using its index?

We can access the characters in a string by referring to its index number inside square brackets []

.

Q28. Write a code to get the desired output of the following

```
string = "Big Data iNeuron"  
desired_output = "iNeuron"
```

```
string[9:16]
```

Q29. Write a code to get the desired output of the following

```
string = "Big Data iNeuron"  
desired_output = "norueNi"
```

```
string[15:8:-1]
```

Q30. Reverse the string given in the above question.

Ans - `string[::-1]`

Q31. How can you delete entire string at once?

Ans - We can delete the entire string at once by using del keyword.

```
str1 = "Hello"
```

```
del(str1)
```

Q32. What is escape sequence?

Ans - The "backslash ()" character as an escape character. In other words, it has a special meaning when we use it inside the strings. As the name suggests, the escape character escapes the characters in a string for a brief moment to introduce unique inclusion.

Q33. How can you print the below string?

'iNeuron's Big Data Course'

ANS - 'iNeuron's Big Data Course'

Q34. What is a list in Python?

ANS - Python list are dynamically sized array, declared in languages like C++ and Java. A list is a collection of things, enclosed in [] and separated by commas

Q35. How can you create a list in Python?

ANS - You can create a list by opening and closing the square brackets

Q36. How can we access the elements in a list?

ANS - We can access the elements in a list by indexing.

Q37. Write a code to access the word "iNeuron" from the given list.

```
lst = [1,2,3,"Hi",[45,54, "iNeuron"], "Big Data"]
```

ANS - `lst = [1,2,3,"Hi",[45,54, "iNeuron"], "Big Data"]`

```
lst[4][2]
```

Q38. Take a list as an input from the user and find the length of the list.

ANS - `n = input("Enter number of elements seprated by space: "). split(" ")`

```
print(len(n))
```

Q39. Add the word "Big" in the 3rd index of the given list.

```
lst = ["Welcome", "to", "Data", "course"]
```

Ans - `lst = ["Welcome", "to", "Data", "course"]`

```
lst.insert(2, "Big")
```

Q40. What is a tuple? How is it different from list?

Ans - Tuple is a collection of Python objects much like a list. The sequence of values stored in a tuple can be of any type, and they are indexed by integers. Tuples are immutable where as list are mutable. We can also faster through the tuples than the list.

Q41. How can you create a tuple in Python?

We can create tuple using round brackets ().

Q42. Create a tuple and try to add your name in the tuple. Are you able to do it? Support your answer with reason.

Ans - No, I can't as tuples are immutable. The work around is it typecast tuple to list and then append.

```
tup = ()
```

```
tup = list(tup)
```

```
tup.append("Vishal")
```

```
tup = tuple(tup)
```

```
tup
```

Q43. Can two tuple be appended. If yes, write a code for it. If not, why?

ANS - Yes, we can

```
tup1 = ("Vishal ")
```

```
tup2 = ("Singh")
```

```
tup1+tup2
```

Q44. Take a tuple as an input and print the count of elements in it.

```
x = input("Enter the values separeted by space: ").split(" ")
```

```
x = tuple(x)
```

```
print(len(x))
```

Q45. What are sets in Python?

ANS - A set is an unordered collection of data types that is iterable, mutable and has no duplicate elements. The order of elements in a set is undefined though it may consist of various elements.

Q46. How can you create a set?

ANS - We can create set using curly brackets {}. Keep in mind empty {} will result in dictionary hence there must be some value in the brackets.

Q47. Create a set and add "iNeuron" in your set.

ANS - set1 = {"iNeuron"}

```
set1
```

Q48. Try to add multiple values using add() function.

Ans - set1.add("Big")

```
set1.add("Data")
```

```
set1
```

Q49. How is update() different from add()?

Ans - We can add more than one element in a single go using update(), but using add() it's not possible.

Q50. What is clear() in sets?

Ans- To remove all the elements from the set, clear() function is used.

Q51. What is frozen set?

Ans- Frozen sets in Python are immutable objects that only support methods and operators that produce a result without affecting the frozen set or sets to which they are applied. While elements of a set can be modified at any time, elements of the frozen set remain the same after creation.

Q52. How is frozen set different from set?

- Ans - Frozen sets are immutable where as sets are mutable.
- Sets can't be used as keys in dictionary where as frozen sets can be used.

Q53. What is union() in sets? Explain via code.

Python set Union() Method returns a new set which contains all the items from the original set.

```
set1 = {2, 4, 5, 6}
```

```
set2 = {4, 6, 7, 8}
```

```
set3 = {7, 8, 9, 10}
```

```
print("set1 U set2 : ", set1 | set2)
```

```
print("set1 U set2 U set3 :", set1 | set2 | set3)
```

Q54. What is intersection() in sets? Explain via code.

Python set intersection() method returns a new set with an element that is common to all set

```
set1 = {2, 4, 5, 6}
```

```
set2 = {4, 6, 7, 8}
```

```
set3 = {4, 6, 8}
```

```
print("set1 intersection set2 : ", set1.intersection(set2))
```

```
print("set1 intersection set2 intersection set3 :", set1.intersection(set2, set3))
```

Q55. What is dictionary in Python?

Dictionary in Python is a collection of keys values, used to store data values like a map, which, unlike other data types which hold only a single value as an element.

Q56. How is dictionary different from all other data structures.

Ans - Dictionary is having key and value pair where as all other data structures have only values in them.

Q57. How can we declare a dictionary in Python?

Ans - We can create dictionary using curly brackets {}.

Q58. What will the output of the following?

```
var = {}  
print(type(var))
```

Ans - dict

Q59. How can we add an element in a dictionary?

Ans - Dict = {}

Dict[0] = "Hello"

Dict[1] = "Course: ["Data Science", "Big Data"]"

Q60. Create a dictionary and access all the values in that dictionary.

Ans - Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}

for i, j in Dict.items():

print(f"Key is {i} and value is {j}")

Q61. Create a nested dictionary and access all the element in the inner dictionary.

ANS - Dict = {"Name": {"f_name": "Vishal", "l_name": "Singh"}, "Experience": 3, "Organisation": "iNeuron"}

for i, j in Dict["Name"].items():

```
print(f'Key is {i} and value is {j}')
```

Q62. What is the use of get() function?

ANS - get() is also to access the elements in dictionary.

```
Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}  
print(Dict.get("Name"))
```

Q63. What is the use of items() function?

Ans - items() method is used to return the list with all dictionary keys with values.

```
Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}  
print(Dict.items())
```

Q64. What is the use of pop() function?

```
Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}  
print(Dict.pop("Name"))
```

Q65. What is the use of popitem() function?

popitem() method removes the last inserted key-value pair from the dictionary and returns it as a tuple.

```
Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}  
print(Dict.popitem())
```

Q66. What is the use of keys() function?

keys() method returns a view object that displays a list of all the keys in the dictionary.

```
Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}
```

```
print(Dict.keys())
```

Q67. What is the use of values() function?

values() is an inbuilt method in Python programming language that returns a view object. The view object contains the values of the dictionary, as a list.

```
Dict = {"Name": "Vishal", "Experience": 3, "Organisation": "iNeuron"}
```

```
print(Dict.values())
```

Q68. What are loops in Python?

Ans - Loops are used to iterate over a block of statement multiple times.

Q69. How many type of loop are there in Python?

Ans - There is for and while loop in Python

Q70. What is the difference between for and while loops?

Ans - When we know the exact number of iterations, we can use for loop. When we want to run till a certain condition is true we can use while loop.

Q71. What is the use of continue statement?

Ans - Continue Statement skips the execution of the program block from after the continue statement and forces the control to start the next iteration.

Q72. What is the use of break statement?

Ans - break statement in Python is used to bring the control out of the loop when some external condition is triggered. break statement is put inside the loop body

Q73. What is the use of pass statement?

Ans - The pass statement is a null statement. But the difference between pass and comment is that comment is ignored by the interpreter whereas pass is not ignored.

Q74. What is the use of range() function?

Ans - range() function returns a sequence of numbers, in a given range. The most common use of it is to iterate sequence on a sequence of numbers.

Q75. How can you loop over a dictionary?

Ans - statesAndCapitals = {


```

        'Karnataka': 'Bengaluru',
        'Maharashtra': 'Mumbai',
        'Rajasthan': 'Jaipur',
        'Bihar': 'Patna'
    }

    for state in statesAndCapitals:

        print(state)

```

Coding problems

Q76. Write a Python program to find the factorial of a given number.

Ans - def factorial(n):

```

    if n < 0:

```

```

        return 0

```

```

    elif n == 0 or n == 1:

```

```

        return 1

```

```

    else:

```

```

        fact = 1

```

```

        while(n>1):

```

```

            fact *= n

```

```

            n -= 1

```

```

        return fact

```

```

n=6

```

```

print(f'Factorial of {n} is {factorial(n)}')

```

Q77. Write a Python program to calculate the simple interest. Formula to calculate simple interest is $SI = (PRT)/100$

Ans - def SI(p,r,t):

si = (p*r*t)/100

print(f'Simple interest is {si}')

return si

SI(8, 8, 6)

Q78. Write a Python program to calculate the compound interest. Formula of compound interest is $A = P(1 + R/100)^t$.

Ans - def CI(p, r, t):

amount = p*(1+r/100)**t

ci = amount - p

print(f'Compound interest is {ci}')

return ci

CI(10000, 10.25, 5)

Q79. Write a Python program to check if a number is prime or not.

Ans - from math import sqrt

def is_prime(n):

prime_flag = 0

if(n > 1):

for i in range(2, int(sqrt(n)) + 1):

if (n % i == 0):

prime_flag = 1

break

if (prime_flag == 0):

```
print(f'{n} is a prime number.')
```

```
else:
```

```
print(f'{n} is not a prime number.')
```

```
else:
```

```
print(f'{n} is not a prime number.')
```

```
is_prime(134)
```

Q80. Write a Python program to check Armstrong Number.

Ans - def check_armstrong(n):

```
s = n
```

```
b = len(str(n))
```

```
sum1 = 0
```

```
while n != 0:
```

```
    r = n % 10
```

```
    sum1 = sum1+(r**b)
```

```
    n = n//10
```

```
if s == sum1:
```

```
    print(f'The given number {s} is armstrong number')
```

```
else:
```

```
    print(f'The given number {s} is not armstrong number')
```

```
check_armstrong(153)
```

Q81. Write a Python program to find the n-th Fibonacci Number.

Ans -

```
def Fibonacci(n):  
    if n <= 0:  
        print("Incorrect input")  
    elif n == 1:  
        return 0  
    elif n == 2:  
        return 1  
    else:  
        return Fibonacci(n-1)+Fibonacci(n-2)  
  
print(Fibonacci(7))
```

Q82. Write a Python program to interchange the first and last element in a list.

```
Ans - def swap_list(newList):  
    size = len(newList)  
    temp = newList[0]  
    newList[0] = newList[size - 1]  
    newList[size - 1] = temp  
    return newList  
  
newList = [15, 12, 35, 17, 9, 56, 29]  
print(swap_list(newList))
```

```
def swap_list(newList):  
    newList[0], newList[-1] = newList[-1], newList[0]  
    return newList  
  
newList = [15, 12, 35, 17, 9, 56, 29]  
print(swap_list(newList))
```

Q83. Write a Python program to swap two elements in a list.

Ans - def swapPositions(list, pos1, pos2):

```
    list[pos1], list[pos2] = list[pos2], list[pos1]  
    return list
```

```
List = [15, 12, 35, 17, 9, 56, 29]  
pos1, pos2 = 1, 3
```

```
print(f"Original list: {List}")  
print(f"Swapped list: {swapPositions(List, pos1, pos2)}")
```

Q84. Write a Python program to find N largest element from a list.

Ans - def n_max_elements(list1, N):

```
    final_list = []  
    for i in range(0, N):  
max1 = 0  
        for j in range(len(list1)):  
            if list1[j] > max1:
```

```

        max1 = list1[j];

        list1.remove(max1);

        final_list.append(max1)

    print(final_list)

list1 = [2, 6, 41, 85, 0, 3, 7, 6, 10]

N = 3

n_max_elements(list1, N)

```

Q85. Write a Python program to find cumulative sum of a list.

Ans - def cumulative_sum(lists):

```

    cu_list = []

    length = len(lists)

    cu_list = [sum(lists[0:x:1]) for x in range(0, length+1)]

    return cu_list[1:]

lists = [10, 20, 30, 40, 50]

print(f'Cumulative sum of the list is {cumulative_sum(lists)}')

```

Q86. Write a Python program to check if a string is palindrome or not.

Ans -

def isPalindrome(s):

```

    if s == s[::-1]:

        return f'{s} is palindrome'

    return f'{s} is not palindrome'

```

```
s = "dad"
isPalindrome(s)
```

Q87. Write a Python program to remove i'th element from a string.

Ans - def remove_ith_element(i):

```
    str1 = "Big Data Bootcamp"
```

```
    str2 = ""
```

```
    for n in range(len(str1)):
```

```
        if n == i:
```

```
            continue
```

```
        else:
```

```
            str2 = str2 + str1[n]
```

```
    return str2
```

```
remove_ith_element(5)
```

Q88. Write a Python program to check if a substring is present in a given string.

```
def check_substring(s2, s1):
```

```
    if (s2.count(s1) > 0):
```

```
        print(f"{s1}" is a substring of "{s2}")
```

```
    else:
```

```
        print(f"{s1}" is not a substring of "{s2}")
```

```
s2 = "Welcome to iNeuron Big Data Bootcamp"
s1 = "iNeuron"
check_substring(s2, s1)
```

Q89. Write a Python program to find words which are greater than given length k.

Ans -

```
def string_greater_than_k(k, str):
```

```
    string = []
    text = str.split(" ")
    for x in text:
        if len(x) > k:
            string.append(x)
    return string
```

```
k = 3
```

```
str="Big Data Bootcamp"
```

```
print(string_greater_than_k(k, str))
```

Q90. Write a Python program to extract unquie dictionary values.

Ans -

```
test_dict = {'iNeuron': [5, 6, 7, 8],
```

```
            'is': [10, 11, 7, 5],
```

```
            'best': [6, 12, 10, 8],
```

```
            'for': [1, 2, 5],
```

```
            'big data': [2, 7, 12, 9]
```

```
        }
```

```
print("The original dictionary is : " + str(test_dict))
```



```
res = list(sorted({ele for val in test_dict.values() for ele in val}))
print("The unique values list is : " + str(res))
```

Q91. Write a Python program to merge two dictionary.

```
Ans - def Merge(dict1, dict2):
    return(dict2.update(dict1))

dict1 = {'a': 10, 'b': 8}
dict2 = {'d': 6, 'c': 4}
print(Merge(dict1, dict2))
print(dict2)
```

Q92. Write a Python program to convert a list of tuples into dictionary.

```
Input : [('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]
Output : {'Sachin': 10, 'MSD': 7, 'Kohli': 18, 'Rohit': 45}
```

```
Ans - print (dict([('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]))
```

Q93. Write a Python program to create a list of tuples from given list having number and its cube in each tuple.

```
Input: list = [9, 5, 6]
Output: [(9, 729), (5, 125), (6, 216)]
```

```
Ans - Input: list = [9, 5, 6]
```

```
Output: [(9, 729), (5, 125), (6, 216)]
```

Q94. Write a Python program to get all combinations of 2 tuples.

```
Input : test_tuple1 = (7, 2), test_tuple2 = (7, 8)
Output : [(7, 7), (7, 8), (2, 7), (2, 8), (7, 7), (7, 2), (8, 7), (8, 2)]
```

```

Ans - test_tuple1 = (7, 2)
test_tuple2 = (7, 8)
res = [(a, b) for a in test_tuple1 for b in test_tuple2]
res = res + [(a, b) for a in test_tuple2 for b in test_tuple1]
print("The filtered tuple : ", str(res))

```

Q95. Write a Python program to sort a list of tuples by second item.

```

Input : [('for', 24), ('Geeks', 8), ('Geeks', 30)]
Output : [('Geeks', 8), ('for', 24), ('Geeks', 30)]

```

```

Ans : def Sort_Tuple(tup):

```

```

    lst = len(tup)

```

```

    for i in range(0, lst):

```

```

        for j in range(0, lst-i-1):

```

```

            if (tup[j][1] > tup[j + 1][1]):

```

```

                temp = tup[j]

```

```

                tup[j]= tup[j + 1]

```

```

                tup[j + 1]= temp

```

```

    return tup

```

```

tup =[('452', 10), ('256', 5), ('100', 20), ('135', 15)]

```

```

print(Sort_Tuple(tup))

```

Q96. Write a python program to print below pattern.

```

*
```

```

* *
* * *
* * * *
* * * * *

```

Ans - def pypart(n):

```

    for i in range(0, n):
        for j in range(0, i+1):
            print("* ",end="")
        print("\r")

```

n = 5

pypart(n)

Q97. Write a python program to print below pattern.

```

*
**
***
****
*****

```

Ans - def inverse_pattern():

```

n=5;i=0
while(i<=n):
    print(" " * (n - i) + "*" * i)
    i+=1

```

inverse_pattern()

Q98. Write a python program to print below pattern.

```

*
* *
* * *
* * * *
* * * * *

```

Ans -

def triangle(n):

```

k = n - 1

for i in range(0, n):
    for j in range(0, k):
        print(end=" ")

    k = k - 1

    for j in range(0, i+1):
        print("* ", end="")

    print("\r")

n = 5

triangle(n)

```

Q99. Write a python program to print below pattern.

```

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5

rows = 5
for i in range(1, rows + 1):
    for j in range(1, i + 1):
        print(j, end=' ')
    print("")

```

Ans -

```

def numpat(n):
    num = 1

    for i in range(0, n):
        num = 1

        for j in range(0, i+1):
            print(num, end=" ")

```

```
        num = num + 1

    print("\r")
```

```
n = 5
```

```
numpat(n)
```

Q100. Write a python program to print below pattern.

```
A
B B
C C C
D D D D
E E E E E
```

Ans -

```
def alphapat(n):
```

```
    num = 65
```

```
    for i in range(0, n):
```

```
        for j in range(0, i+1):
```

```
            ch = chr(num)
```

```
            print(ch, end=" ")
```

```
        num = num + 1
```

```
    print("\r")
```

```
n = 5
```

```
alphapat(n)
```