## Assignment Part-1

Q1. Why do we call Python a general purpose and high-level programming language?

ANS: Python is used in almost all the fields for automation, healthcare, industries etc. It is a solution for many different problems and not just a single issue. It is high level because it is user-friendly compared to other programming languages such as Java, C, C++ etc.

Q2. Why is Python called a dynamically typed language?

ANS: Variables in python are automatically determined at the run time and its type of variables doesn't need to be explicitly mentioned unlike C++, C.

Q3. List some pros and cons of the Python programming language?

ANS: PROS:

1. User friendly

2. Object oriented

3. Versatile usage

4. Wide varieties of libraries available for all the software engineering fields

CONS:

1. As it is a interpreter language it executes the code line by line making the execution speed slow

Q4. In what all domains can we use Python?

ANS: 1. Healthcare

2. Game development

3. Web development

4. BigData

5. Machine Learning and AI

Q5. What are variables and how can we declare them?

ANs: Variables are just names for the location which holds the values in the memory. In python we can just declare variables using the "=" operator.

EX: a = 1 (Value of a holds int type)

a = "Kannappan" (Value of a holds string type variable)

a = [1,2,3] (Value of a holds array type)

Q6. How can we take an input from the user in Python?

ANS: Using input() method

Q7. What is the default datatype of the value that has been taken as an input using input() function?

AnS: Str

Q8. What is type casting?

AND: Converting the type of one variable to another is called as type casting

EX: a = "1"

print(type(a)) #op=str

a = int(1)

print(type(a)) # op=int

Q9. Can we take more than one input from the user using a single input() function? If yes, how? If not, why?

ANS: Yes we can take more than one input using a split() function.

EX: a,b,c = input("Enter 3 values").split()

print(a + "-" + b + "-" +c)

OP: Enter 3 values:

1 2 3

Code\_res:

1-2-3

Q10. What are keywords?

ANS: Keywords are special words which have specific meaning and purpose. Ex: For, if, while, and, or etc.

Q11. Can we use keywords as a variable? Support your answer with reason.

ANS: No we can use keywords as variables as it by default has some predefined meaning allocated.

Q12. What is indentation? What's the use of indentation in Python?

ANS: Python uses indentation to depict a block of code. For other programming languages we use "{}" to indicate the block of code but for python the correct space at the starting of every line i.e indentation indicates the block of code.

Q13. How can we throw some output in Python?

ANS: Output we can get it or verified using print statements. In python functions we can use return statement to throw an output

Q14. What are operators in Python?

ANS: Operators are used to perform some form of actions with the variables used in the program.

EX: Arithmetic operator, binary operator, comparison operator etc

Q15. What is the difference between / and // operators?

ANS: / -> It does the division with the decimal point result

// - It does the division without decimal point result and get the value as a whole number

Q16. Write a code that gives the following as an output.

```

iNeuroniNeuroniNeuroniNeuron

```

ANS: # iNeuroniNeuroniNeuroniNeuron

a = int(input("Enter number of times iNeuron needs to be printer: "))

print(a)

res = ""

for i in range(0,a):

res+= "iNeuron"

print(res)

Q17. Write a code to take a number as an input from the user and check if the number is odd or even.

ANS: num = int(input("Enter a number:"))

if (num%2==0):

print(str(num)+" is even")

else:

print(str(num)+ " is odd")

Q18. What are boolean operators?

ANS: and, or and not are known as boolean operators.

Q19. What will be the output of the following?

```

1 or 0

0 and 0

True and False and True

1 or 0 or 0

```

ANS:

1 or 0 => 1

0 and 0 => 0

True and False and True => False

1 or 0 or 0 =? 1

Q20. What are conditional statements in Python?

ANS: ==, !=, <=,<,>=,> these are known as conditional operators

Q21. What is the use of 'if', 'elif' and 'else' keywords?

ANS: To check the conditions we use if, elif and else statements. For making a decision we usually use these clauses.

elif is used for nested if else conditions.

Q22. Write a code to take the age of the person as an input and if age >= 18 display "I can vote". If the age is < 18, it displays "I can't vote".

ANS:

age = int(input("Enter your age:"))

if (age >= 18):

print("I can vote")

else:

print("I can't vote")

Q23. Write a code that displays the sum of all the even numbers from the given list.

```

numbers = [12, 75, 150, 180, 145, 525, 50]

```

ANS:

```

numbers = [12, 75, 150, 180, 145, 525, 50]

define allowEvenNumbers(lst):

even\_lst=[]

for num in lst:

if num%2==0:

even\_lst.append(num)

return even\_lst

sum = 0

numbers\_lst = numbers

evenNumList=allowEvenNumbers(numbers\_lst)

for each\_number in evenNumList:

sum=sum+each\_number

print(sum)

```

Q24. Write a code to take 3 numbers as an input from the user and display the greatest no as output.

ANS:

num1, num2, num3 = input("Enter 3 numbers separated by space:").split()

max = max(int(num1), int(num2), int(num3))

print(max)

Q25. Write a program to display only those numbers from a list that satisfy the following conditions

- The number must be divisible by five

- If the number is greater than 150, then skip it and move to the next number

- If the number is greater than 500, then stop the loop

```

numbers = [12, 75, 150, 180, 145, 525, 50]

```

ANS:

op\_lst = []

numbers = [12, 750, 150, 180, 145, 525, 50]

num\_lst = numbers

for each\_num in num\_lst:

if each\_num > 500:

break

elif each\_num > 150:

continue

else:

if each\_num%5 == 0:

op\_lst.append(each\_num)

print(op\_lst)

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

ANS: Sequences of characters within a single or double quotes are called strings.

EX: name = “Kannappan”

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

ANS: we can access strings like arrays using indexes.

EX: name = “Kannappan”

print(name[1])

OP: ‘a’

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

```

string = "Big Data iNeuron"

desired\_output = "iNeuron"

```

CODE:

string = "Big Data iNeuron"

lst\_str = string.split()

print(lst\_str[-1])

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

```

string = "Big Data iNeuron"

desired\_output = "norueGa"

```

Code:

string = "Big Data iNeuron"

lst\_str = string.split()

lst\_wrd = lst\_str[-1]

print(lst\_wrd[::-1])

Q30. Reverse the string given in the above question.

ANS: We can do string reversal in multiple ways but python allows us to do easily using lst\_wrd[::-1]

Q31. How can you delete an entire string at once?

CODE:

string = "Big Data iNeuron"

del(string)

print(string)

Q32. What is an escape sequence?

ANS: It is a sequence of characters used within a string that doesn’t represent itself and it will represent some other characters.

For ex: To print the output in a new line we will use \n

Q33. How can you print the below string?

```

'iNeuron's Big Data Course'

```

CODE:

**print('iNeuron\'s Big Data Course')**

Q34. What is a list in Python?

ANS: List is used to store multiple elements in a single variable

Q35. How can you create a list in Python?

ANS: List is created using square brackets

EX: lst = [] -> creates a empty list

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

ANS: Elements within the list are also accessed using index.

EX: lst = [1,2,3,4]

print(lst[3])

OP: 4

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

```

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

```

CODE:

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

print(lst[4][-1])

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

CODE:

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

print(len(lst))

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

```

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

```

CODE:

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

lst.insert(2, "Big")

print(lst)

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

Tuple also has multiple elements in a single variable. The difference between tuple and list is tuple is immutable and list is mutable. Tupple is enclosed with normal brackets and List is enclosed within square brackets

Q41. How can you create a tuple in Python?

CODE:

tup = ()

print(tup)

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: After creating a tuple it is not possible to add any elements to it, as it is unchangeable and it also doesn’t have any functions like append to do the job.

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

ANS: As tuples are immutable it is not possible to do as given. But we can have some workaround like converting a tuple into a list and using the extend function to append and again convert it back to tuple.

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

CODE:

tup = (2,"kannappan",9)

print(len(tup))

Q45. What are sets in Python?

ANS: Sets are also the same as that of list and tuple. We can have multiple elements in a single variable. Elements are enclosed within curly braces

Q46. How can you create a set?

CODE:

set\_1 = {2,"kannappan",9}

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

CODE:

set\_1 = {2,"kannappan",9}

set\_1.add("iNeuron")

print(set\_1)

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

ANS: add function accepts only one argument. So we can add multiple elements to a set. If we use update function to a existing set then it is possible to add multiple elements

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

ANS:Update() function allows iterable values to add to a set but add() function allows only one value to be added

Q50. What is clear() in sets?

Clear function will delete all the elements within the set.

Q51. What is a frozen set?

ANS: It is an immutable set. No duplicate values will also be present.

Q52. How is a frozen set different from a set?

ANS: Immutable sets are called frozenset. Once we have a frozenset we cannot edit or add its values. Mainly used to keep the Keys of the dictionary static.

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

ANS: Union is the same as the append function for list. It combines two sets and exclude duplicate values

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

ANS: Intersection function return only the common elements between two sets

Q55. What is a dictionary in Python?

ANS: Dictionary is one of the primarily used data types in python. Mainly used to store the data in the from of key value pairs

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

ANS: It stores the data in the form of Key value pairs.

Q57. How can we declare a dictionary in Python?

ANS:

Dict = {“Name”: “Kannappan”}

print(Dict[‘Name’])

op= Kannappan

Q58. What will be the output of the following?

```

var = {}

print(type(var))

```

ANS: dict

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

ANS: dict[“Key”] = Value

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

CODE:

my\_dict = {'name': 'Kannappan', 'age': 27}

print(my\_dict['age'])

Output: 27

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

ANS: A dictionary within another dictionary is called a nested dictionary.

CODE:

dict = {"name":"Kannappan", "age": 27, "marks":{"Eng":100,"Tamil":100,"Maths":98}}

print(dict["marks"])

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

ANS: get() returns the value of the key passed within the function

dict = {"name":"Kannappan", "age": 27, "marks":{"Eng":100,"Tamil":100,"Maths":98}}

print(dict.get("marks"))

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

ANS: items function will display key value pairs of a dictionary into tuples within a list.

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

ANS: Pop function removes the key and its values passed within the function.

Q65. What is the use of the pop items() function?

ANS: Popitem() will remove the last element from the dictionary

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

ANS: keys() will return only the keys from a dictionary

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

ANS: Values() will return all the values from the dictionary in the form of tuples within a list

Q68. What are loops in Python?

ANS: Iterating over the elements present within a list / dictionary /sets are called loops

Q69. How many types of loops are there in Python?

ANS: We have for loops and while loops in python

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

ANS: For loop is used when the number of conditions is known and while loop will be executed until the condition is true and exits when condition fails

Q71. What is the use of the continue statement?

AND:Continue statement is used to end the current iteration of the loop.

Q72. What is the use of a break statement?

ANS: Break statement is used to end or terminate the entire loop

Q73. What is the use of a pass statement?

ANS: pass statement is just a dummy code inorder to continue the code without error. It's like a placeholder

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

ANS: range function returns an iterative or sequence of numbers.

Q75. How can you loop over a dictionary?

ANS:

d = {"Name": "Kannappan","age":27}

for k,v in d.items():

print(k,v)

### Coding problems

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

CODE:

***n = 5***

***def fact(n):***

***fact = 1***

***for i in range(1,n+1):***

***fact = fact\*i***

***return fact***

***print(fact(n))***

Q77. Write a Python program to calculate the simple interest. Formula to calculate simple interest is SI = (P\*R\*T)/100

CODE:

***def si(p,r,t):***

***return (p\*r\*t)/100***

***simple Interest = si(1000000,20,10)***

***print(simple Interest)***

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

CODE:

***def si(p,r,t):***

***return p\*(pow((1 + r / 100), t))***

***simple Interest = si(1000000,20,10)***

***print(simple Interest)***

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

CODE:

n = 4

***is\_prime = "No"***

***if n == 2 or n ==3:***

***is\_prime = "Yes"***

***else:***

***for i in range(2,(n//2)+1):***

***if n % i == 0:***

***is\_prime = "No"***

***break***

***else:***

***is\_prime = "Yes"***

***print(is\_prime)***

Q80. Write a Python program to check Armstrong Number.

CODE:

***org\_number = 54748***

***number = org\_number***

***sum = 0***

***total\_digit = len(str(number))***

***for i in range(0,total\_digit):***

***each\_digit = number%10***

***digit\_cube = pow(each\_digit,total\_digit)***

***sum = sum+digit\_cube***

***number = number//10***

***if sum == org\_number:***

***print("%d is an armstrong number" %org\_number)***

***else:***

***print("%d is not an armstrong number" %org\_number)***

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

CODE:

***def fib(n):***

***if n == 1:***

***return n***

***else:***

***return n\*fib(n-1)***

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

CODE:

***temp = lst[0]***

***lst[0] = lst[-1]***

***lst[-1] = temp***

***print(lst)***

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

CODE:

Same code provided in the above problem can be used to do this operation

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

CODE:

***max(lst)***

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

***CODE:***

***sum(lst)***

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

CODE:

***str = "MoM"***

***op = (str == str[::-1])***

***if op == True:***

***print("%s is a palindrome" %str)***

***else:***

***print("%s is not a palindrome" %str)***

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

CODE:

***print(str[:i]+str[i+1:])***

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

CODE:

***main\_str = "Kannappan is a genius"***

***sub\_str = "geniu"***

***lst\_str = main\_str.split()***

***if sub\_str in lst\_str:***

***print("{0} word is present in the statement {1}".format(sub\_str,main\_str))***

***else:***

***print("{0} word is not present in the statement {1}".format(sub\_str,main\_str))***

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

CODE:

***lst\_str = ["I","am","a","genius",".","I","am","from","India"]***

***k = 4***

***nw\_lst = []***

***for word in lst\_str:***

***word\_length = len(word)***

***if word\_length > k:***

***nw\_lst.append(word)***

***print(", ".join(nw\_lst))***

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

CODE:

***org\_dict = {'A': [10, 35, 15, 4], 'B': [400, 69, 8, 10], 'C': [69, 12, 400, 8], 'D': [35, 77, 21]}***

***lst = []***

***for i in org\_dict.values():***

***lst***

***print(set(lst))***

Q91. Write a Python program to merge two dictionaries.

CODE:

***dict1 = {"a":1,"b":2}***

***dict2 = {"c":3,"d":4}***

***dict1.update(dict2)***

***print(dict1)***

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

```

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

Output : {'Sachin': 10, 'MSD': 7, 'Kohli': 18, 'Rohit': 45}

```

CODE:

***lst\_value = [('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]***

***dict = {}***

***for tup in lst\_value:***

***dict[tup[0]] = tup[1]***

***print(dict)***

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

```

Input: list = [9, 5, 6]

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

```

CODE:

***lst = [9, 5, 6]***

***lst\_op = []***

***for i in lst:***

***tup = (i,pow(i,3))***

***lst\_op.append(tup)***

***print(lst\_op)***

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)]

```

CODE:

***test\_tuple1 = (7, 2)***

***test\_tuple2 = (7, 8)***

***lst\_op = []***

***for i in test\_tuple1:***

***for j in test\_tuple2:***

***lst\_op.append((i,j))***

***for i in test\_tuple2:***

***for j in test\_tuple1:***

***lst\_op.append((i,j))***

***print(lst\_op)***

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)]

```

CODE:

***lst\_tuple = [('for', 24), ('Geeks', 8), ('Geeks', 30)]***

***lst\_tuple.sort(key=lambda a: a[1])***

***print(lst\_tuple)***

Q96. Write a python program to print the pattern below.

```

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

```

CODE:

for i in range(0, 6):

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

print("\* ",end="")

print("\r")

Q97. Write a python program to print the pattern below.

```

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

```

CODE:

***rows = 6***

***for i in range(0,rows):***

***for j in range(0, rows-i):***

***print(" ", end=" ")***

***for k in range(0,i):***

***print("\*", end=" ")***

***print()***

Q98. Write a python program to print the pattern below.

```

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

```

CODE:

***rows = 6***

***for i in range(0,rows):***

***for j in range(0, rows-i):***

***print("", end=" ")***

***for k in range(0,i):***

***print("\*", end=" ")***

***print()***

Q99. Write a python program to print the pattern below.

```

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

```

CODE:

***for i in range(1,6):***

***num = 1***

***for j in range(0,i):***

***print(num, end = " ")***

***num+=1***

***print()***

Q100. Write a python program to print the pattern below.

```

A

B B

C C C

D D D D

E E E E E

```

CODE:

***for i in range(0,5):***

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

***print(chr(65+i), end = " ")***

***print()***