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

Ans) Because it is easy for humans to understand

Q2. Why is Python called a dynamically typed language?

Ans) Python don't have any problem even if we don't declare the type of variable. It states the kind of variable in the runtime of the program. Python also take cares of the memory management which is crucial in programming. So, Python is a dynamically typed language

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

Ans) Pros- Easy to Read, Learn and Write ,Improved Productivity , Interpreted Language , Dynamically Typed , Free and Open-Source , Vast Libraries Support, Portability

Cons- Slow Speed , Not Memory Efficient , Weak in Mobile Computing , Database Access ,Runtime Errors.

Q4. In what all domains can we use Python?

Ans)Python is the go-to programming language for domains such as artificial intelligence, machine learning and deep learning, it's no surprise that it's also a fundamental tool for any data scientist.

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

Ans. Variable is name given to a specific memory location , variable should start with alphabet.

Ex. abc\_123

\_123

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

Ans) Input is taken from the user taking input function

Ex. a = input(“Your age:”)

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

Ans) The default datatype of the value is always string

Q8. What is type casting?

Ans) Changing a datatype of a value is known as type casting.

Q9. Can we take more than one input from the user using single input() function? If yes, how? If no, why

Ans) Yes , using split() function

Ex. { a , b , c = input(“Enter 3 Numbers”),split() }

Q10. What are keywords?

Ans) Keywords are some predefined and reserved words in python that have special meanings. Keywords are used to define the syntax of the coding. The keyword cannot be used as an identifier, function, and variable name. All the keywords in python are written in lower case except True and False

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

Ans) No , Because python doesn’t support this functions

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

Ans) the indentation in Python is very important. Python uses indentation to indicate a block of code.

Q13. How can we throw some output in Python?

Ans) We throw output using print function

Ex. print(“Hi , iNeuron”)

Q14. What are operators in Python?

Ans) In Python, there are seven different types of operators: arithmetic operators, assignment operators, comparison operators, logical operators, identity operators, membership operators, and boolean operators

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

Ans) / denotes division operators for float values

// denotes division operators for int values

Q16

name = "iNeuron"

print(name\*4)

Q17

a = int(input("Please provide a number:"))

print(a)

if (a % 2 == 0):

print("The Number is Even")

else :

print("The Number is odd")

Q18. What are boolean operator?

Ans) In Python, the Boolean type is bool , which is a subtype of int . Boolean values are the values True or False (with a capital T and F) in Python. A Boolean variable is a variable that can be either True or False .

Q19. What will the output of the following?

```

1 or 0

0 and 0

True and False and True

1 or 0 or 0

```

Ans) 1

0

False

1

Q20. What are conditional statements in Python?

Ans) The if statement is a conditional statement in python, that is used to determine whether a block of code will be executed or not. Meaning if the program finds the condition defined in the if statement to be true, it will go ahead and execute the code block inside the if statement.

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

Ans) if… elif…else are conditional statements that provide you with the decision making that is required when you want to execute code based on a particular condition. The if… elif…else statement used in Python helps automate that decision making process.

Q22

age = int(input("Please provide your age:"))

if (age >= 18):

print("I can vote")

else :

print("I can't vote")

Q23

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

sum = 0

for i in range(7):

if (numbers[i] % 2 == 0):

sum = sum + numbers[i]

print(sum)

Q24

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

if (a > b and a > c):

print("A is greatest")

elif (b > a and b > c):

print("B is greatest")

else:

print("C is greatest")

Q25

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

b = []

for i in a:

if i > 150:

if i > 500:

break

continue

if i % 5 == 0:

b.append(i)

print(b)

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

Strings can be created by enclosing characters inside a single quote or double-quotes. Even triple quotes can be used in Python but generally used to represent multiline strings and docstrings.

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

Individual characters in a string can be accessed by specifying the string name followed by a number in square brackets ( [] ). String indexing in Python is zero-based: the first character in the string has index 0 , the next has index 1 , and so on.

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

string = "Big Data iNeuron"

print("Desired\_output",string[9:16])

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

string = "Big Data iNeuron"

print("Desired\_output",string[-1:-8:-1])

Q30. Resverse the string given in the above question.

string = "Big Data iNeuron"

print("Desired\_output",string[-1::-1])

Q31. How can you delete entire string at once?

string = "Big Data iNeuron"

del(string)

Q32. What is escape sequence?

Character combinations consisting of a backslash (\) followed by a letter or by a combination of digits are called "escape sequences." To represent a newline character, single quotation mark, or certain other characters in a character constant, you must use escape sequences

Q33. How can you print the below string?

string = "'iNeuron's Big Data Course'"

print(string)

Q34. What is a list in Python?

Lists are used to store multiple items in a single variable. Lists are one of 4 built-in data types in Python used to store collections of data, the other 3 are Tuple, Set, and Dictionary, all with different qualities and usage.

Q35. How can you create a list in Python?

New\_list = [123,"PSP", 12.22]

print(type(New\_list))

print(New\_list)

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

New\_list = [123,"PSP", 12.22]

print(type(New\_list))

print(New\_list)

print(New\_list[0])

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

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

print(lst[4][2])

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

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"]

lst.insert(2,"Big")

print(lst)

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

The primary difference between tuples and lists is that tuples are immutable as opposed to lists which are mutable. Therefore, it is possible to change a list but not a tuple. The contents of a tuple cannot change once they have been created in Python due to the immutability of tuples

Q41. How can you create a tuple in Python?

my\_tuple = (1,2,3)

print(type(my\_tuple))

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

You can't add elements to a tuple because of their immutable property.

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

my\_tuple = (1,2,3)

s = (4,5,6)

op = my\_tuple + s

print(op)

print(type(my\_tuple))

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

my\_tuple = (1,2,3)

print(len(my\_tuple))

Q45. What are sets in Python?

Sets are used to store multiple items in a single variable. Set is one of 4 built-in data types in Python used to store collections of data, the other 3 are List, Tuple, and Dictionary, all with different qualities and usage. A set is a collection which is unordered, unchangeable\*, and unindexed.

Q46. How can you create a set?

A set is created by placing all the items (elements) inside curly braces {} , separated by comma, or by using the built-in set() function.

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

my\_set = {}

print(type(my\_set))

my\_set = set()

print(type(my\_set))

my\_set.add("iNeuron")

print(my\_set)

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

my\_set = {}

print(type(my\_set))

my\_set = set()

print(type(my\_set))

my\_set.add("iNeuron")

a = {"is" , "best"}

my\_set.update(a)

print(my\_set)

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

add method directly adds elements to the set while the update method converts first argument into set then it adds the list is hashable therefore we cannot add a hashable list to unhashable set.

Q50. What is clear() in sets?

The clear() method removes all elements in a set.

Q51. What is frozen set?

The frozenset() function returns an immutable frozenset object initialized with elements from the given iterable.

Q52. How is frozen set different from set?

Frozen set is just an immutable version of a [Python set](https://www.programiz.com/python-programming/set) object. While elements of a set can be modified at any time, elements of the frozen set remain the same after creation.

Due to this, frozen sets can be used as keys in [Dictionary](https://www.programiz.com/python-programming/dictionary) or as elements of another set. But like sets, it is not ordered (the elements can be set at any index).

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

a = {1 , 2 , 3 , 4}

b = {2 , 3, 4 , 5 ,6}

print("AUB=", a.union(b))

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

a = {1 , 2 , 3 , 4}

b = {2 , 3, 4 , 5 ,6}

print("AUB=", a.intersection(b))

Q55. What is dictionary ibn Python?

Dictionaries are used to store data values in key:value pairs. A dictionary is a collection which is ordered\*, changeable and do not allow duplicates. As of Python version 3.7, dictionaries are ordered.

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

List and tuple is an ordered collection of items. Dictionary is unordered collection. List and dictionary objects are mutable i.e. it is possible to add new item or delete and item from it. Tuple is an immutable object

Q57. How can we delare a dictionary in Python?

var = {}

print(type(var))

Q58. What will the output of the following?

<class 'dict'>

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

devBio = {

  "name": "Ihechikara",

  "age": 120,

  "language": "JavaScript"

}

devBio["role"] = "Developer"

print(devBio)

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

thisdict = {

  "brand": "Ford",

  "model": "Mustang",

  "year": 1964

}

print(thisdict["model"])

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

people = {1: {'name': 'John', 'age': '27', 'sex': 'Male'},

          2: {'name': 'Marie', 'age': '22', 'sex': 'Female'}}

print(people[1]['name'])

print(people[1]['age'])

print(people[1]['sex'])

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

Does the same work of accessing element.

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

The items() method returns a view object. The view object contains the key-value pairs of the dictionary, as tuples in a list.

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

The pop() method removes the specified item from the dictionary. The value of the removed item is the return value of the pop() method

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

The popitem() method removes the item that was last inserted into the dictionary. In versions before 3.7, the popitem() method removes a random item. The removed item is the return value of the popitem() method

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

To get the keys from the dictionary.

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

To get the values from the dictionary.

Q68. What are loops in Python?

A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string). This is less like the for keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages.

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

There are mainly 2 types of loops  
For loop  
While Loop

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

he for loop is used when we already know the number of iterations, which means when we know how many times a statement has to be executed. That is why we have to specify the ending point in the for loop initialization. When we need to end the loop on a condition other than the number of times, we use a while loop

Q71. What is the use of continue statement?

The continue statement is used to skip the rest of the code inside a loop for the current iteration only. Loop does not terminate but continues on with the next iteration

Q72. What is the use of break statement?

The break statement terminates the loop containing it. Control of the program flows to the statement immediately after the body of the loop.

Q73. What is the use of pass statement?

In Python programming, the pass statement is a null statement which can be used as a placeholder for future code.

Suppose we have a loop or a function that is not implemented yet, but we want to implement it in the future. In such cases, we can use the pass statement.

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

The range() function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and stops before a specified number.

Q75. How can you loop over a dictionary?

You can loop through a dictionary by using a for loop. When looping through a dictionary, the return value are the keys of the dictionary, but there are methods to return the values as well.

CODING QUESTIONS

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

n = int (input ("Enter a number: "))

factorial = 1

if n >= 1:

    for i in range (1, n+1):

        factorial = factorial \*i

print ( "Factorial of the given number is:", factorial)

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

p = int( input("Enter the value:"))

r = int( input("Enter the value:"))

t = int( input("Enter the value:"))

S = p \* r \* t / 100

print("Simple Interest is",S)

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

P = int( input("Enter the value:"))

R = int( input("Enter the value:"))

t = int( input("Enter the value:"))

c = int(1+ R/100)

d = int(c^t)

e = P \* d

print("Simple Interest is",e)

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

num = int(input("Please enter a number"))

if num > 1:

   # check for factors

   for i in range(2,num):

       if (num % i) == 0:

           print(num,"is not a prime number")

           print(i,"times",num//i,"is",num)

           break

   else:

       print(num,"is a prime number")

Q80. Write a Python program to check Armstrong Number.

num = int(input("Please enter a number"))

# Changed num variable to string,

# and calculated the length (number of digits)

order = len(str(num))

# initialize sum

sum = 0

# find the sum of the cube of each digit

temp = num

while temp > 0:

   digit = temp % 10

   sum += digit \*\* order

   temp //= 10

# display the result

if num == sum:

   print(num,"is an Armstrong number")

else:

   print(num,"is not an Armstrong number")

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

def fibonacci(n):

    a = 0

    b = 1

    if n < 0:

        print("Incorrect input")

    elif n == 0:

        return a

    elif n == 1:

        return b

    else:

        for i in range(2, n):

            c = a + b

            a = b

            b = c

        return b

print(fibonacci(9))

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

def swapList(list):

    first = list.pop(0)

    last = list.pop(-1)

    list.insert(0, last)

    list.append(first)

    return list

li = [1, 9, 2, 10, 19, 30]

print(li)

print("Swapped list: ",swapList(li))

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

def swapPositions(list, pos1, pos2):

    first = list.pop(pos1)

    second = list.pop(pos2-1)

    list.insert(pos1, second)

    list.insert(pos2, first)

    return list

List = [2, 6, 12, 23, 8, 9]

pos1, pos2  = 1, 3

print("Earlier list: ",List)

print("Swapped list: ",swapPositions(List, pos1-1, pos2-1))

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

def N\_max\_elements(list, N):

    result\_list = []

    for i in range(0, N):

        maximum = 0

        for j in range(len(list)):

            if list[j] > maximum:

                maximum = list[j]

        list.remove(maximum)

        result\_list.append(maximum)

    return result\_list

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

N = 2

print(N, "max elements in ",list1)

print(N\_max\_elements(list1, N))

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

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

new\_list=[]

j=0

for i in range(0,len(list)):

    j+=list[i]

    new\_list.append(j)

print(new\_list)

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

def isPalindrome(s):

    return s == s[::-1]

s = input("Enter your name:")

ans = isPalindrome(s)

if ans:

    print("Yes")

else:

    print("No")

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

test\_str = "GeeksForGeeks"

new\_str = ""

for i in range(len(test\_str)):

    if i != 2:

        new\_str = new\_str + test\_str[i]

print ("The string after removal of i'th character : " + new\_str)

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

MyString1 =input("Enter a string")

if "need" in MyString1:

    print("Yes! it is present in the string")

else:

    print("No! it is not present")

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

def string\_k(k, str):

    string = []

    text = str.split(" ")

    for x in text:

        if len(x) > k:

            string.append(x)

    return string

k = 3

str ="geek for geeks"

print(string\_k(k, str))

Q90. Write a Python program to extract unquire dictionary values

test\_dict = {'gfg' : [5, 6, 7, 8],

            'is' : [10, 11, 7, 5],

            'best' : [6, 12, 10, 8],

            'for' : [1, 2, 5]}

print("The original dictionary is : " + str(test\_dict))

x=list(test\_dict.values())

y=[]

res=[]

for i in x:

    y.extend(i)

for i in y:

    if i not in res:

        res.append(i)

res.sort()

print("The unique values list is : " + str(res))

Q91. Write a Python program to merge two dictionary.

dict\_1 = {1: 'a', 2: 'b'}

dict\_2 = {2: 'c', 4: 'd'}

dict\_3 = dict\_2.copy()

dict\_3.update(dict\_1)

print(dict\_3)

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

list\_1=[('Sachin', 10), ('MSD', 7), ('Kohli', 18), ('Rohit', 45)]

dict\_1=dict()

for student,score in list\_1:

    dict\_1.setdefault(student, []).append(score)

print(dict\_1)

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

def cubeoflist(li):

    result=[(num, num\*\*3) for num in li]

    return result

li = [9, 5, 6]

print(cubeoflist(li))

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

test\_tuple1 = (7, 2)

test\_tuple2 = (7, 8)

print("The original tuple 1 : " + str(test\_tuple1))

print("The original tuple 2 : " + str(test\_tuple2))

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.

tuple = [('for', 24), ('Geeks', 8), ('Geeks', 30)]

print("Orignal Tuple List :" ,tuple)

def Sort(tuple):

    tuple.sort(key = lambda a: a[1])

    return tuple

print("Sorted Tuple List:" ,Sort(tuple))

Q96. Write a python program to print below pattern.

def pypart(n):

    if n==0:

        return

    else:

        pypart(n-1)

        print("\* "\*n)

n = 5

pypart(n)

Q97. Write a python program to print below pattern

height = 5

for row in range(1, height+ 1):

    print(" " \* (height - row) +"\*" \* row)

Write a python program to print below pattern.

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)

Write a python program to print below pattern.

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.

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)