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

**-->**

Python is a general-purpose language because it is used to solve variety of problem statements and is used in wide range of applications. It is versatile in nature.

It can be used is machine learning, web development, software testing, data analysis, scripting.

Python is considered as high-level programming language because it is easily understandable by humans. Python is far from machine code and closer to natural language like English, which in turn makes it easy for any novice beginner easy to grasp the concept. Hence, programming is easier in python.

**Q2. Why is Python called a dynamically typed language?**

**-->**

Python is called dynamically typed because same variable can have different type at different times during execution. In other word, the interpreter in python does not assign a type to the variable because the type can change at runtime.

e.g.: var\_a = 34

print(type(var\_a)) --> outputs type as 'int'

var\_a = "My code"

print(type(var\_a)) --> outputs type as 'string'

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

**-->**

Pros:

1. Easy to use: Python's syntax very similar to English. Python requires less lines of code as compared to other programming languages like C, C++, etc

2. Productive language: because its easily implementable, it can be applied in many domains quickly and hence increases productivity.

3. Interpreted language: The python code is executed directly during runtime (without any delay in compiling the code first). Python executes the code directly one code line after another. The debugging in python occurs at runtime.

4. Open source: easily available for free and easily distributable

Cons:

1. Slow Speed: Since python is dynamically typed and is interpreted, the code lines are executed one by one and hence takes more time to execute when the code is too long

2. Inefficient memory consumption

3. Since python doesn't follow many syntax rules, sometimes the length of code can become too large.

**Q4. In what all domains can we use Python?**

**-->**

Python can be used in: Data science, Machine learning, Statistics, Cybersecurity, Game development, Back-end web application development, Embedded applications

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

**-->**

Variables in python are basically name given to a memory location. Variable is a container for storing the any data values

Variable declaration e.g.

x = "I am learning along with INeuron"

print(type(x)) -- > string

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

**-->**

first\_name\_var = input ("Please enter your first name: ")

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

**-->**

the default datatype of the value that has been taken as an input using input () function is string

**Q8. What is type casting?**

**-->**

Type casting in python refers to changing the datatype of a certain variable

# int variable

x = 10

# typecast to float

n = float(a)

print(n)

print(type(n))

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

**-->**

Yes, we can take more than one input from the user using single input() function :-- by using split() method

e.g:

# Taking multiple inputs at a time

x = [int(x) for x in input("Enter multiple values: ").split()]

print("Number of list is: ", x)

**Q10. What are keywords?**

**-->**

Keywords are specially reserved words which possesses some specific meaning and uses. Keywords cannot be used for any purpose other than its defined purpose.

e.g if, else, yield, break, assert, etc

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

**-->**

No keywords cannot be used as a variable. This is because the purpose of keywords in python is already defined and allocated . User cannot change its functionality in any way

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

**-->**

Indentation means spaces at the beginning of the code line in python. Basically, indentation is a way of telling a Python interpreter that the group of statements belongs to a particular block of code. A block is a combination of all these statements.

**Q13. How can we throw some output in Python?**

**-->**

Using print() function.

e.g

print("My name is xyz")

**Q14. What are operators in Python?**

**-->**

Operators are symbols in python which have their designated functionality of computation. e.g +, -, /, //, %, \*\*, etc

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

**-->**

/ Refers to standard divison

The result of standard division (/) is always a float, even if the dividend is evenly divisible by the divisor

e.g. 10 / 5

2.0

type(10 / 5)

<class 'float'>

// refers to floor division . Result (Quotient is always rounded to the next smallest integer)

e.g.

>>> 10 // 4

2

>>> 10 // -4

-3

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

**```**

**iNeuroniNeuroniNeuroniNeuron**

**```**

**-->**

inp\_str = "iNeuron"

print(inp\_str\*4)

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

**-->**

number = input("Enter the number = ")

if int(number) % 2 == 0:

print(f"Number {number} is even")

else:

print(f"Number {number} is odd")

**Q18. What are Boolean operator?**

**-->**

Boolean operators are the operators whose output when examined is either True or False.

e.g. and, or, not, ==, >, <, !=, etc

**Q19. What will the output of the following?**

**```**

**1 or 0**

**0 and 0**

**True and False and True**

**1 or 0 or 0**

**-->**

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?**

**-->**

Conditional Statements allows for conditional execution of a statement or group of statements based on the value of an expression.

e.g. if clause, else clause, elif clause, nested if clauses, etc

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

**-->**

if, elif and else keywords are decision making keywords. It will evaluate test expressions and gives result only if the condition is true otherwise gives remaining output.

e.g.

num = 3.4

if num > 0:

print("Positive number")

elif num == 0:

print("Zero")

else:

print("Negative number")

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

**-->**

age= input("Enter the age of a person = ")

if int(age) >= 18:

print(f"I can vote")

else:

print(f"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]**

**-->**

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

even\_number\_list = []

for i, num in enumerate(numbers):

if num % 2 == 0:

even\_number\_list.append(num)

sum\_even\_num = sum(even\_number\_list)

print(f"Sum of all even numbers : {sum\_even\_num}")

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

**-->**

list\_input\_num = [int(x) for x in input("Enter all three inputs which are separated by a space: ").split()]

greatest\_num = max(list\_input\_num)

print(f"Greatest number from {list\_input\_num} is {greatest\_num}")

**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]**

**-->**

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

output\_num\_list = []

for ele in numbers:

if (ele % 5 == 0) :

if ele > 500:

break

if ele > 150:

continue

output\_num\_list.append(ele)

print(output\_num\_list)