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

- Python is a general-purpose language, it can be used to create a variety of different programs and can be used for any specific problems.
- Python is a high-level programming language, it's easy for humans to understand.

Q2. Why is Python called a dynamically typed language?

- In python it is not necessary to declare the type of the variable. It declares the type of the variable during the runtime of the program.
- Python also takes care of memory management. So, Python is a dynamically typed language.

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

Pros:

- Python is easy to learn and read.
- Python improves the productivity.

Cons:

- Python has a large collection of libraries.
- Pyhon is open-source and has large community.

Q4. In what all domains can we use Python?

- Python is used in various domains like Macine Learning, Web Development, Artificial Intelligence, Gaming, Data science, Scientific Programming etc,.

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

A variable is a name given to an object or funtion or class in python. To define a variable, we must follow the below properties.

- A variable name must start with a letter or the underscore character.
- A variable name cannot start with a number.
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _).
- Variable names are case-sensitive (name, Name and NAME are three different variables).
- The reserved words(keywords) cannot be used naming the variable.

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

- To take an input from the user we have to delare as "int(input("Enter data."))".

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

- string is the defult datatype that willbe taken as input by using the input() function.

Q8. What is type casting?

- converting the type of data by using type casting.
- There are two types Implicit(conversion of data types is done automatically by Python) and Explicit type casting(conversion of data types is done by the user).

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

- Yes, it is possible to take more than one input using python.
- eg: x = list(map(int, input("Enter multiple values: "). split()))

Q10. What are keywords?

- Python keyword is specially reserved words with specific meanings and purposes.
- They are different from built-in functions and types. Keywords are the building blocks of any python program.

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

- Keywords are used to define the syntax of the coding. The keyword cannot be used as an identifier, function, and variable name.

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

- Indentation refers to the spaces at the beginning of a code line. Where in other programming languages the indentation in code is for readability only,
- 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?

- By using the print() function we can get the output in python

Q14. What are operators in Python?

- An Operator is used toperform operations. There are different types of operators,
- Arithmatic, Assignment, Bitwise, logical, binary, unary, comparison, Identity, Membership operators.

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

- / is regular division(returns float) and // is floor division(returns int).
- Floor division was introduced in python 3.

eg:

x = 5/2 => 2.5

y = 5//2 => 2

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

```iNeuroniNeuroniNeuroniNeuron```

```
x = "iNeuroniNeuroniNeuroniNeuron"
print(x)
```

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

```
x = int(input("Enter the value:"))
if x%2==0:
    print("Even Number.")
else:
    print("Odd Number.")
```

Q18. What are boolean operator?

- Boolean operators are used to perform boolean operations.
- Boolean or logical operators are AND, OR, and NOT.
- The keywords and, or and not are the Python operators for these operations.

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  output: 1

0 and 0  output: 0

True and False and True  output: False

1 or 0 or 0  output: 1

Q20. What are conditional statements in Python?

- The if, elif and else statements are conditional statements in python, that is used to determine whether a block of code will be executed or not.

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

- If-elif-else statement is used in Python for decision-making i.e the program will evaluate test expression and
- will execute the remaining statements only if the given test expression turns out to be true.
- This allows validation for multiple expressions.

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 = int(input("Enter 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]```

for i in numbers:

```
evenSum = 0
 if i%2==0:
            evenSum = evenSum +i
print("The sum of all even numbers is:", evenSum)
```

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

```
x =list(map(int,input(“Enter numbers:- “)).splitt())
x.sort()
print(x[-1])
```

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]
```

```
for i in numbers:
    if i > 500:
        break
    elif i > 150:
        continue
    elif i % 5 == 0:
        print(i)
```