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

Ans: Python is not written in a machine-readable language, and it's very easy for humans to understand, this is why it is a general purpose language.
Python is not a straight compiled language like Java or C, but an interpreted dynamic language that has to be run in the given system using another program instead of it's local processor. When compiled, other languages turn into assembly and run directly in the processor. Hence, being an interpreted language, which is not subject to the processor, makes Python a high-level programming language.

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 takes 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 of Python programming language:

1. Beginner friendly, easy to learn
2. Highly scalable
3. Extensive libraries
4. Flexible and extensible
5. Large community

Cons of Python programming language:

1. Python is slow compared to other compiled languages, as it requires more computational power.
2. Python language comes with high memory usage, and the high memory consumption should be tackled carefully during the project.
3. Multithreading in Python is not exactly true due to its global interpreter lock (GIL). The multithreading model in Python does not have thread that run at the same time. Here, one thread can hold the GIL at one time, which clearly implies that this is not the exact multithreading.
4. Python maximizes a lot of errors and bugs in the system, as it is dynamically typed language.
5. Python is unsuitable for complicated designs.

Q4. In what all domains can we use Python?

Ans: We can use Python in different domains like, Data Science, Artificial Intelligence, Machine Learning, Automation, Application Development, Console Applications, Desktop GUI, Audio/Video Applications etc.

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

Ans: Variable is a name given to a specific memory location, a simple declaration and assignment of a variable is like, a = 5

We need to keep a few rules in mind regarding a variable declaration and naming conventions:

1. Variable name can't start with digit, it should always start with alphabet.
2. We can't use any special symbol in a variable except '_' underscore. Ex: abc_123 is a correct name, whereas abc@123 is not.
3. Variable name can also start with underscore. Ex: _abc is correct name.

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

Ans: We can take user input in Python using input() function. We can also pass argument to this function for better user experience. For example:

1. name = input()
2. name = input("Please enter your name:")

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 that has been taken as an input using input() functions is 'String'.

Q8. What is type casting?

Ans: Type casting is the method to convert the variable data type into a certain data type in order to the operation required to be performed by users. Ex: int_var = int(12.98)

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

Ans: Yes, we can take more than one input from the user using single input() function by using split() function along with that.

```
Ex: num1, num2, num3 = input("Please enter 3 numbers respectively:").split()
    print(num1, num2, num3)
```

Q10. What are keywords?

Ans: Keywords are 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. Ex: if, elif, else, break, continue, for, while, etc.

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

Ans: No, we can't use keywords as a variable.

As keywords are specially reserved words with specific meanings and purposes. And they can't be used for any purposes other than the one they have made for. If you try assigning any value to keywords, you will get syntax error.

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

Ans: Indentation refers to the spaces in 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. Python will give error if you skip an indentation in code.

Q13. How can we throw some output in Python?

Ans: We can throw output in Python using print() function. Ex: print("Hello World"), print("This is an integer variable:", int_var), etc.

Q14. What are operators in Python?

Ans: Operators are special symbols in Python that carry out arithmetic or logical computation. There are a few operators available in Python like, arithmetic operators, logical operators, conditional operators, bitwise operators, assignment operators, etc. And the value that the operator operates on is called the Operand. Ex: 2 + 3, here '+' is arithmetic operator and 2 and 3 are operand.

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

Ans: '/' is known as float-division in Python. Ex: 5 / 2 = 2.5
'//' is known as integer-division or floor-division. Ex: 5 // 2 = floor(2.5) = 2

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

Ans:

```
# Python 3
# Output Required: iNeuroniNeuroniNeuroniNeuron

org_name = 'iNeuron'
org_name = org_name * 4
print(org_name)
```

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

Ans:

```
# Python 3
# Take input of a number from user and check the number is odd or even

int_var = int(input("Please enter any integer number:"))    # type casted to integer
if int_var % 2 == 0:
   print("Even Number!")
else:
   print("Odd Number!")
```

Q18. What are boolean operator?

Ans: Logical operators like, ‘and’, ‘or’, ‘not’ are also knows as boolean operator in Python.
If both the operand is True then ‘and’ operator will give ‘True’ as a result.
If any one of the operand is True then ‘or’ operator will give ‘True’ as a result.
And ‘not’ operator will just reverse the result, if the operand says True, it will give result as ‘False’ and vice versa.

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 or 0 will give 1 as output.
0 and 0 will give 0 as output.
True and False and True will give False as output.
1 or 0 or 0 will give 1 as output.

Q20. What are conditional statements in Python?

Ans: A conditional statement as the name suggests itself, is used to handle conditions in your program. These statements guide the program while making decisions based on the conditions encountered by the program.

There are 3 types of conditional statements in Python.

1. if statement
2. if-else statement
3. If-elif-else statement

1. <u>if statement:</u>

```
if int_var > 8:
    print(“Number is greater than 8”)
```

If the ‘int_var’ variable value is greater than 8, then the respective print statement will be executed.

2. if-else statement:

```
if int_var > 8:
    print(“Number is greater than 8”)
else:
    print(“Number is less than 8”)
```

If the ‘int_var’ variable value is greater than 8, then the print statement inside the if will be executed, otherwise the print statement inside the else block will be executed.

3. if-elif-else statement:

```
if int_var > 8:
    print(“Number is greater than 8”)
elif int_var == 8:
    print(“Number is equals to 8)
else:
    print(“Number is less than 8”)
```

If the ‘int_var’ variable value is greater than 8, then the print statement inside the if will be executed, else if the ‘int_var’ value is equals to 8, then the print statement inside the elif block will be executed, otherwise the print statement inside the else block will be executed.

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

Ans: There are 3 types of conditional statements in Python.

4. if statement
5. if-else statement
6. If-elif-else statement

1. if statement:

```
if int_var > 8:
    print(“Number is greater than 8”)
```

If the ‘int_var’ variable value is greater than 8, then the respective print statement will be executed.

2. if-else statement:

```
if int_var > 8:
    print(“Number is greater than 8”)
else:
    print(“Number is less than 8”)
```

If the ‘int_var’ variable value is greater than 8, then the print statement inside the if will be executed, otherwise the print statement inside the else block will be executed.

3. if-elif-else statement:

```
if int_var > 8:
    print(“Number is greater than 8”)
```

```
    elif int_var == 8:
        print(“Number is equals to 8)
    else:
        print(“Number is less than 8”)
```

If the ‘int_var’ variable value is greater than 8, then the print statement inside the if will be executed, else if the ‘int_var’ value is equals to 8, then the print statement inside the elif block will be executed, otherwise the print statement inside the else block will be executed.

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".

Ans:

```
# Python 3
# Take age as user input, and check whether it is >= 18 or not, if so print "I can
vote", otherwise if < 18, print "I can't vote"

int_age = int(input("Please enter your age:"))
official_vote_age = 18

if int_age >= official_vote_age:
   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:

```
# Python 3
# 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_sum = 0

for i in range(len(numbers)):
   if numbers[i] % 2 == 0:
       even_sum = even_sum + numbers[i]

print(even_sum)
```

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

Ans:

```
# Python 3
# Take 3 numbers as an input from the user and display the greatest no as output.

num1 = int(input("Please enter the first number:"))
num2 = int(input("Please enter the second number:"))
num3 = int(input("Please enter the third number:"))

greatest_number = 0

if num1 > num2 and num2 > num3:
   greatest_number = num1
elif num2 > num1 and num1> num3:
   greatest_number = num2
else:
   greatest_number = num3

print(greatest_number)
```

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:

```
# Python 3
# Display only those numbers from the 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]
conditional_list = []

for i in range(len(numbers)):
   if numbers[i] <= 500:
       if numbers[i] <= 150:
           if numbers[i] % 5 == 0:
               conditional_list.append(numbers[i])
   else:
```

```
        break

# print(conditional_list)
for i in range(len(conditional_list)):
    print(conditional_list[i])
```