Assignment Part-1

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

Python is an object-oriented, high-level programming language. Object-oriented means this language is based around objects (such as data) rather than functions, and high-level means it's easy for humans to understand.

Q2. Why is Python called a dynamically typed language?

Python is both a strongly typed and a dynamically typed language. Strong typing means that variables do have a type and that the type matters when performing operations on a variable. Dynamic typing means that the type of the variable is determined only during runtime.

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

Pros	Cons
Beginner-friendly	Issues with design
Large Community	Slower than compiled languages
Flexible and Extensible	Security
Extensive Libraries	Work Environment

Embeddable	High memory consumption
Highly Scalable	Dynamically-typed language
IoT Opportunities	Complex multithreading
Portable	Garbage collection leads to potential memory losses

Q4. In what all domains can we use Python?

Desktop GUI

- Machine learning / Artificial intelligence
- Data analytics and data visualization
- Web development
- Game development
- Mobile app development
- Embedded systems

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

Python variables are of four different types: Integer, Long Integer, Float, and String. Integers are used to define numeric values; Long Integers are used for defining integers with bigger lengths than a normal Integer

There is no binding in Python to declare variables before we use it. We also need not explicitly declare variables with their data type. When we assign a value to Python variables, they are automatically declared

We use the “=” operator for value assignment.

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

Taking input is a way of interact with users, or get data to provide some result. Python provides two built-in

methods to read the data from the keyboard. These methods are given below.

- input(prompt)
- raw_input(prompt)

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

By default, input returns a string. So the name and age will be stored as strings

Q8. What is type casting?

The conversion of one data type into the other data type is known as type casting in python or type conversion in python. Python supports a wide variety of functions or methods like: int(), float(), str(), ord(), hex(), oct(), tuple(), set(), list(), dict(), etc. for the type casting in python

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

split( ) function helps us get multiple inputs from the user and assign them to the respective variables in one line

Q10. What are keywords?

In Python, there are approximately around thirty-three (33) keywords, and a few of the keywords generally used in the program coding are break, continue, true, false, and, or, not, for, while, def, class, if, else, elif, import, from, except, exec, print, return, yield, lambda, global, etc.

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

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?

Sometimes you want Python to throw a custom exception for error handling. You can do this by checking a condition and raising the exception, if the condition is True. The raised exception typically warns the user or the calling application.

You use the “raise” keyword to throw a Python exception manually. You can also add a message to describe the exception

Q14. What are operators in Python?
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 the difference between / and // operators?

/ is regular division(returns float) and // is floor division(returns int).

Floor division was introduced in python 3.

eg:

```
1. x = 5/2 #2.5
2. y = 5//2 #2
```

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

iNeuroniNeuroniNeuroniNeuron
```
print('NeuroniNeuroniNeuroniNeuron')

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

# Python program to check if the input number is odd or even.
# A number is even if division by 2 gives a remainder of 0.
# If the remainder is 1, it is an odd number.

num = int(input("Enter a number: "))
if (num % 2) == 0:
   print("{0} is Even".format(num))
else:
   print("{0} is Odd".format(num))


Q18. What are boolean operator?

Boolean Operators are simple words (AND, OR, NOT or AND NOT) used as conjunctions to combine or exclude keywords in a search

Q19. What will the output of the following?
```
1 or 0

0 and 0

True and False and True

1 or 0 or 0
```

Zero is used to represent false, and One is used to represent true
In Python 3. x True and False are keywords and will always be equal to 1 and 0

Q20. What are conditional statements in Python?

- if statements.
```

- if-else statements.
- elif statements.

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

It allows us to check for multiple expressions. If the condition for if is False , it checks the condition of the next elif block and so on. If all the conditions are False , the body of else is 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".

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]
```

```
list1 = [12, 75, 150, 180, 145, 525, 50]

# iterating each number in list
for num in list1:

    # checking condition
    if num % 2 == 0:
        print(num, end=" ")
```

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

```
num1 = int(input())
num2 = int(input())
num3 = int(input())
print(max(num1,num2,num3),"is the greatest")  # Built-in function "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]
```

```
numbers = [12, 75, 150, 180, 145, 525, 50]
# iterate each item of a list
for item in numbers:
    if item > 500:
        break
    elif item > 150:
        continue
    # check if number is divisible by 5
    elif item % 5 == 0:
        print(item)
```