**Q1. Python as a general purpose and high-level programming language --**

Python is called as high-level programming language because it is “easy to understand by humans and also it is in a human readable format”.

**Q2. Python called a dynamically typed language**

Suppose if we don’t declare the type of variable in python it doesn’t have any problem. It states the type during the run time of the program, and it takes the responsibility of memory management. So, it is called as dynamically typed language.

**Q3. Pros & Cons of Python**

|  |  |
| --- | --- |
| **Pros** | **Cons** |
| * Free & Open-source license * Easy to use, understand, and beginner friendly. * Straight-forward syntax. * Programmer doesn’t need to worry about declaring the variables and their data types * **Larger libraries support** - so we can find almost all the functions we need and we don’t want to depend on other external libraries. | * Slow execution speed. * Uses the large amount of memory. * Slow processing power so python is weak in mobile computing. * Since python is a dynamically typed language it occurs run-time errors. * Testing in python takes more time compared with other languages. |

**Q4.Domains we use python**

Python is used in multiple domains such as **Data scientist, Machine learning, deep learning, networking, artificial intelligence…**

**Q5. Declaring variables**

Variable is a name given to a memory location and acts as a container to store values

Declaring variable: first\_name = ‘Aswanth’

**Q6.Taking input from user**

first\_name = input (“Enter your first name: “)

second\_name = input (“Enter your second name: “)

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

**STRING Data type**

**Q8.Type casting -** The conversion of one data type into another data type is known as type casting.

The type casting is of 2 types

**Implicit** – In this, python converts data type into another data type automatically.

#Implicit type casting

# automatically converts a to int

a = 4

print(type(a)) # int

# automatically converts

# b to float

b = 2.5

print(type(b))  #<class 'float'>

# automatically converts c to float as it is float addition

c = a + b

print("c =",c)

print(type(c))  #<class 'float'>

**Explicit –** In this , the user converts the data type into another data type.

#Explicit type casting

num\_int = 10

num\_str = '25'

print(type(num\_int))    #<class 'int'>

print(type(num\_str))    #<class 'str'>

casted\_num\_str = int(num\_str)

total\_sum = num\_int + casted\_num\_str

print("total sum = ",total\_sum) #total sum =  35

print(type(total\_sum))  #<class 'int'>

#example-2

a = 3.6

#typecast of float to int

b =  int(a)

print("b = ",b) #b =  3

print("type of b : ",b) #type of b :  3

**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 from the user. It is possible by using the split() function method

#using split() fncn

x,y,z = input("Enter three values \_\_ \_\_ \_\_ ").split()   #Enter three values \_\_ \_\_ \_\_ VIRAT KOHLI CRICKETER

print("Enter your first name :",x)  #Enter your first name : VIRAT

print("Enter your second name :",y) #Enter your second name : KOHLI

print("Enter your Profession:",z)   #Enter your Profession: CRICKETER

#OUTPUT :

# Enter three values \_\_ \_\_ \_\_ VIRAT KOHLI CRICKETER

# Enter your first name : VIRAT

# Enter your second name : KOHLI

# Enter your Profession: CRICKETER

**Q10. keywords**

Keywords are pre-defined reserved words that have a special meaning to the compiler. We cannot use a keyword as a variable name or function names. Some of the keywords are (True, False, and, Break, Continue)

**Q11**. **Can we use keywords as a variable?**

Keywords cannot be used as a variable name because they are used to define the syntax of coding.

and = 6

print(and) #SyntaxError: invalid syntax

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

Indentation in Python refers to adding white spaces before the statement to a particular block of code and it is used to indicate the block of code.

**Q13.Throwing output in python-** We throw output by using the **print( )** function.

**Q14.Operators in python:**

Python operators are used to perform the operations on variables and values. Operators are divided into,

Arithmetic - used with numeric values to perform common mathematical operations. (+,-,\*,%,/,//)

Assignment – assigning the values to variables (x=5, y += 10 => y = y+10)

Comparison – compares the two values and returns Boolean value(x = = y, a < b, c > d, y != z)

Logical – used to check whether the condition is True or False (and, or, not )

#Arithmetic Operators

a = 10

b = 20

print("Addition of a and b : ",a+b)

print("Subtraction of a and b : ",a-b)

print("Multiplication of a and b : ",a\*b)

print("Division of a and b : ",a/b)

print("Modulo of a and b : ",a%b)

print("Floor Division of a and b : ",a//b)

#Assignment operator

a = 5

b = a + 5 # 5+5 =10

b+=a # b=b+a => 10+5 = 15

print("value of a and  b : ",a,b)

#comparison operator

a = 10

b = 3

print(a>b) # True

#Logical operator

x = 50

y = 80

print((x>y)and(y<x)) #False

print((x<y)or(x>y)) #True

print(not(x>y)) #True

**Q15.** **difference between / and // operators**

/ - Division operator which performs division.

# / **Division Operation**

a= 47

b = 4

print("Division of a and b : ",a/b)  #Division of a and b :  11.75

**#Floor division**

print("Floor Division of a and b : ",a//b)  #Floor Division of a and b :  11

// - floor division gives the quotient value as rounded , if the quotient is positive (eg.2.57)it gives the value as 2, suppose it is negative (eg,-2.56) it gives as -3.

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

**```iNeuroniNeuroniNeuroniNeuron**

str\_1 = 'iNeuron'

str\_2 = 4 \* str\_1

print(str\_2)

Q**17. Write a code to take a number as an input from the user and check if the number is odd or even.**

#get no. from user

num\_1 = int(input("Entera a number : "))

#check no. is odd or even

if (num\_1%2 == 0):

    print("The number is EVEN")

else:

    print("The number is ODD")

**Q18. What are Boolean operator?**

Boolean operator used to check the conditions whether True or False by using and, or, not

b = 10

c = 50

print((b>c)and(c<b)) #False

print((b<c)or(b>c)) #True

print(not(b>c)) #True

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

```

**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 are defined as it tells the computer to execute certain block of code when it is true suppose if the condition is false it executes another block of code.

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

These are the conditional statements used when you want to execute the block of code based on the certain conditions.

**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".**

# take input age of person from user

age = int(input("Enter the 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]

```

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

result = 0

for num in numbers:

  if num%2 == 0:

    result = result+num

  else:

    continue

print("sum of even numbers : ",result)

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

a = int(input("Enter the value of a :"))

b = int(input("Enter the value of b :"))

c = int(input("Enter the value of c :"))

if (a>b) and (a>c):

  print("a is the greatest number")

elif (b>c) and (b>a):

  print("b is the greatest number")

else:

  print("c is the 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]**

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

i = []

for num in numbers:

  if num > 150:

    if num > 500:

      break

  elif num%5==0:

    i.append(num)

print(i)