1) what is a variable in python?

Ans) A variable is an string or a character which takes any values of different data types.

eg: x = 1, str = "Python is a good language", charac = 'a' etc.

2) How do you create a variable?

Ans) We just type a suitable word for value being assigned and use an assignment operator to assign the value to the variable typed.

eg: variable = "new Variable", area = pi*(r**2) etc.

3) How do you check the value within a vbariable?

Ans) We just print the value of the variable on the screen using a print in-built function and passing the variable to the print() function.

```
eg:   area = 24
      print(area)


      output : 24
```

4) How do you create multiple variables in a single statement?

Ans) We create multiple variables in a single statement by using a comma(",") operator.

```
eg: mulvar1, mulvar2, mulvar3, mulvar4 = 4,"variables","are being created",'.'
    print(mulvar1,mulvar2,mulvar3,mulvar4)


    output: 4 variables are being created.
```

5) How do you create multiple variables with same value?

Ans) We create multiple variables with same value by using equals symbol between the multiple bariables and assign value to one of the variables among them.

```
eg: mulvar1,mulvar2,mulvar3,mulvar4 = "Sairam"
    print(mulvar1,mulvar2,mulvar3,mulvar4)


    output: Sairam Sairam Sairam Sairam
```

6) How do you change the value of a variable?

Ans) We can change the value of a variable same like assigning a value to the variable by using an equal to operator, if it is used the value of the variable gets updated automatically.

```
eg: name = "ROHITH"

   print(name)

   name = "KODI ROHITH"

   print(name)


   output : ROHITH

               KODI ROHITH
```

7) How do you reassign a variable by modifying the previous value?

Ans) We reassign the value of variable by using an assignment operator after the variable being assigned some value.

```
eg: val = 23

   print(val)

   val = 54

   print(val)


   output: 23

              54
```

Here the value of the variable 'val' is being reassigned as 54 after the variable being assigned 23 as it's value.

8)What does the statement counter +=4 do?

Ans) val += 4 is a complex statement where the value of the variable 'val' is being incremented by 4, but writing it in simple terms, we can write it as val = val + 4.

9) What are the rules for naming a variable?

Ans) The rules for naming a variable are, the name assigned should be relvent to the value being assigned and should be short and sweet, the variable name must not start with any numbers and special characters, it's better to use an "_" to separate two words instead of writing it in a camel case manner.

10) Are variable names case-sensitive? Do a_variable, A_Varaible, and A_VARIABLE represent the same variable or different ones?

Ans) Yes names of the variables are case-sensitive, the variables - "a_variable", "A_Varibale", "A_VARIABLE" are different from each other and can contain different values.

11) What is Syntax? Why is it important?

Ans) Syntax is nothing but the way you write the code, and the semantics you follow while writing the code, it is very important becase, python is very particular about in and incase if there is any extra space or any other character added where it's not necessary, the it would be difficult for the code to execute, or we may land at error or at wrong results.

12) What happens if you execute a statement with invalid syntax?

Ans) If we execute a statement with an invalid syntax, we land at errors, compilation issues, infinte loops and incorrect or wrong results.

13) How do you check the data type of a variable?

Ans) We check the data type of a variable in python by using a built-in function called "type()", we pass the variable in the type() function and output would be the data type of the variable.

14) What are the built-in data types in python?

Ans) built-in data types in python are as follows:

1) numbers

2) variables

3) Strings

4) Arrays

5) List

6) Tuple

7) Dictionary

8) Set

15) What is a primitive data type?

Ans) A primitive datatype is a standalone datatype.

16) What are the primitive data types available in python?

Ans) Numbers

17) What is a data structure or container data type?

Ans) A data structure is a user made data type which can have a mixture of data types, like a data type which can take all kinds of data.

eg: list, set, tuple, dictionary etc.

18) What are the container types available in python?

Ans) Container types available in python are lists, tuple, dictionary, sets etc.

19) What kind of data does the Integer data type represent?

Ans) Integer represents all positive and negative numbers.

eg: 1,2,3,4,-5,-6,-7 etc...

20) What are the numerical limits of the integer data type?

Ans) No limits of the integer data type, allowed till the memory space of the system.

21) What kind of data does the float data type represent?

Ans) Float data type represent real numbers.

eg: 12.434, 432543.43564365, 0.000432 etc...

22) How does Python decide if a given number is a float or an integer?

Ans) Python decides a given number is a float or an integer depending on their binary representation in the system, floating point numbers have a different representation than integers.

23) How can you create a variable which stores a whole number, e.g., 4 but has the float data type?

Ans) by using the built-in type casting number.

eg: int(float number) - which would convert a float number into integer number.

24) How do you create floats representing very large (e.g., 6.023 x 10^23) or very small numbers (0.000000123)?

Ans) by representing using the scientific notation.

eg: 1.23e-7 for 0.0000123

25) What does the expression `23e-12` represent?

Ans) 23e-12 is represented as 0.00000000023.

26) Can floats be used to store numbers with unlimited precision?

Ans) Yes

27) What are the differences between integers and floats?

Ans) The difference between an integer and float will always result in a float number.

28) How do you convert an integer to a float?

Ans) Use a built-in float() function by passing the integer to the float function which would convert integer to a floating point number.

29) How do you convert a float to an integer?

Ans) Use a built-in int() function by passing the float to the integer function which would convert float to an integer.

30) What is the result obtained when you convert 1.99 to an integer?

Ans) We get 1 when we convert 1.99 to an integer.

eg: int(1.99)

```
  output: 1
```

31) What are the data types of the results of the division operators `/` and `//`?

Ans) When we use"/" - we get a floating point number as the result, but when we use"//" - we get an integer as the result.

32) What kind of data does the Boolean data type represent?

Ans) It represent only two values - "True" or "False".

33) Which types of Python operators return booleans as a result?

Ans) All comparision operators of python return a boolean as a result.

34) What happens if you try to use a boolean in arithmetic operation?

Ans) True acts as 1 and False acts as 0.

35) How can any value in Python be covered to a boolean?

Ans) by using a built-in bool() type-casting function.

eg: bool(variable) - variable gets assigned the value of boolean, either true or false.

36) What are truthy and falsy values?

Ans) The truthy and falsy values are boolean values, which takes values either 0 or 1.

37) What are the values in Python that evaluate to False?

Ans) The value that gets evaluated to false is 0.

38) Give some examples of values that evaluate to True.

Ans) Any valus of a variable gets assigned to be true.

eg: rama = 4, ram_name = "Rama", The variables both rama and ram_name are evaluated to true.

39) What kind of data does the None data type represent?

Ans) variables being generated but not assigned any values are of None data type.

40) What is the purpose of None?

Ans) It indicates that the variable exists but, no value is being assigned to it.

41) What kind of data does the String data type represent?

Ans) String data type represents an entire line as a string in the code.

42) What are the different ways of creating strings in Python?

Ans) The different ways of creating strings in python are by using double quotes and single quotes and using a built-in function and type-cast any data-type to a string.

43) What is the difference between strings creating using single quotes, i.e. `'` and `'` vs. those created using double quotes, i.e. `\"` and `\"`?

Ans) There exists no difference between the strings created using double quotes and single quotes.

44) How do you create multi-line strings in Python?

Ans) By using a triple double quotes or by indicating that the string still continues in the next line.

45) What is the newline character, `\\n`?

Ans) The newline character indicates the compiler that the next ouput must be printed in the next line.

46) What are escaped characters? How are they useful?

Ans) Special escape sequences indicated by a "\" backslash.

47) How do you check the length of a string?

Ans) by using a built-in length function which would return the length of the string.

eg: len(string)

48) How do you convert a string into a list of characters?

Ans) we can do this by many ways, mentioning few, we can do it using a for loop and accessing each character of the string using string indexing and adding it to the list, or using a built-in functions.

49) How do you access a specific character from a string?

Ans) by accessing it using an index of the string or by finding out the character using a built-in function.

50) How do you access a range of characters from a string?

Ans) We access a range of characters from a string using slicing.

eg: a = "Sairam"

   print(a[:])

   output : Sairam

51) How do you check if a specific character occurs in a string?

Ans) we loop through the entire string and find if a specific character exists or use a built-in "in" operator to check whether a specific character exists in the string.

52) How do you check if a smaller string occurs within a bigger string?

Ans) By using a built-in "in" operator to check if the smaller string exists in a bigger string.

53) How do you join two or more strings?

Ans) by using a "+" operator between the strings.

54) What are \"methods\" in Python? How are they different from functions?

Ans) methods are built-in functions which are being used by the user, whereas functions are user-defined and used by the user.

55) What do the `.lower`, `.upper` and `.capitalize` methods on strings do?

Ans) .lower converts all characters of string to lowercase, wheras .upper and .capitalize converts all characters of string to uppercase.

56) How do you replace a specific part of a string with something else?

Ans) by using a built-in replace() method, by passing both the words, the previous word in the string and the new word to be replaced in it.

eg: a = "King"

   a.replace("King","Queen")

57) How do you split the string \"Sun Mon Tue Wed Thu Fri Sat\" into a list of days?

Ans) By assigning a variable the string.split() function.

eg: a = "Sun,Mon,Tue,Wed,Thu,Fri,Sat"

```
   print(a.split())


   output : ['Sun', 'Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat']
```

58) How do you remove whitespace from the beginning and end of a string?

Ans) .strip() method remove the white spaces from the beginning and end of the string.

59) What is the string `.format` method used for? Can you give an example?

Ans) Its is used to replace that part of the string.

```
eg: print("Sairam {} and welcome to the {} class".format("all","python"))


   output : Sairam all and welcome to the python class
```

60) What are the benefits of using the `.format` method instead of string concatenation?

Ans) .format is easier and dynamic in nature where concatenation is not dynamic.

61) How do you convert a value of another type to a string?

Ans) by using a built-in function str(another data type), which will convert the other data type to string.

```
eg: a = 42
   print(str(a))


   output: '42'
```

62) How do you check if two strings have the same value?

Ans) by using a "==" operator between the strings and would return true if they are equal or it would return false.

63) Where can you find the list of all the methods supported by strings?

And) In the directory of strings in the compiler

64) What is a list in Python?

Ans) A list in python can be considered as an array which can take different type values.

65) How do you create a list?

Ans) We can create a list in python by following the below syntax.

eg: list = [1,'two',3.14,0]

66) Can a Python list contain values of different data types?

Ans) Yes, it can have multiple data types values.

67) Can a list contain another list as an element within it?

Ans) Yes, python can have a list an element in another list.

68) Can you create a list without any values?

Ans) Yes we can create a list without any values in python.

69) How do you check the length of a list in Python?

Ans) By using a len(list_name) built-in function.

70) How do you retrieve a value from a list?

Ans) We can retrieve a value from a list in many ways, one using indexing and other can be slicing, or just using a print statement and passing the list variable name in the function which would print the entire function.

71) What is the smallest and largest index you can use to access elements from a list containing five elements?

Ans) The smallest index would be 0 as list indexes start from 0 and the largest would be 4, as the last index would 1 less than the numbers of elements in the list.

72) What happens if you try to access an index equal to or larger than the size of a list?

Ans) Index out if bound error would occur.

eg: IndexError: list index out of range

73) What happens if you try to access a negative index within a list?

Ans) The list accesses the elements from backside, the negative indexing starts with -1.

74) How do you access a range of elements from a list?

Ans) By the specifying the index range while printing the list.

75) How many elements does the list returned by the expression `a_list[2:5]` contain?

Ans) 3 Elements

76) What do the ranges `a_list[:2]` and `a_list[2:]` represent?

Ans) The range a_list[:2] represent all the elements from beginning till the index 1, and a_list[2:] represent all the elements beggining from index 2 and till the end of the list.

77) How do you change the item stored at a specific index within a list?

Ans) We can change the item stored at a specific index using the index number i.e a_list[index_number].

78) How do you insert a new item at the beginning, middle, or end of a list?

Ans) By using a built-in function list.insert(specifying the index and item to be added to the list).

79) How do you remove an item from al list?

Ans) By using a built-in function list.remove(specifying the element to be removed from the list).

eg: list.remove("sairam")

80) How do you remove the item at a given index from a list?

Ans) By using a built-in function list.remove(specifying the element to be removed from the list using indexing).

81) How do you check if a list contains a value?

Ans) By using a contains keyword.

82) How do you combine two or most lists to create a larger list?

Ans) We just add a "+" operator between the lists i.e we concatenate the lists, to form a larger list.

83) How do you create a copy of a list?

Ans) We can create a copy of list by assigning the list to one more other variable to which it has to be copied.

84) Does the expression `a_new_list = a_list` create a copy of the list `a_list`?

Ans) Yes it copies the entire list of the a_list to the a_new_list.

85) Where can you find the list of all the methods supported by lists?

Ans) In the list directory of python.

86) What is a Tuple in Python?

Ans) An immutable list is said to be a tuple where, the data or the elements are stored in a paranthesis.

87) How is a tuple different from a list?

Ans) It is immutable and tuple has paranthesis whereas list has sqaure brackets.

88) Can you add or remove elements in a tuple?

Ans) No.

89) How do you create a tuple with just one element?

Ans) We cannot create a tuple with a single element, it is considered as normal data type.

90) How do you convert a tuple to a list and vice versa?

Ans) By using the built-in type casting function.

91) What are the `count` and `index` method of a Tuple used for?

Ans) Count method is used to count the number of elements in the tuple and index method is used the find out and element using indexing.

92) What is a dictionary in Python?

Ans) A dictionary in python is a type of associative array which has a key and a value.

93) How do you create a dictionary?

Ans) We create a dictionary in the following manner.

eg: person = { "name":"Tillu", "Age":21, "class": "1st Msc"}

94) What are keys and values?

Ans) Keys and values are objects in python dictionary which are key-value pairs.

95) How do you access the value associated with a specific key in a dictionary?

Ans) By specifying the key name in the dictionary.

eg: person["name"]

96) What happens if you try to access the value for a key that doesn't exist in a dictionary?

Ans) It gives a keyerror as follows.

```
eg:    print(person["place"])
          KeyError: 'place'
```

97) What is the `.get` method of a dictionary used for?

Ans) .get method is used to retrieve a value given a key if it exists.

98) How do you change the value associated with a key in a dictionary?

Ans) By the using the key of the value as follows.

eg: person["name"] = "Raj"

99) How do you add or remove a key-value pair in a dictionary?

Ans) We can add or remove a key-value pair in a dictionary by using the built-in functions of the dictionary.

100) How do you access the keys, values, and key-value pairs within a dictionary?

Ans) We can access the keys using the .keys() built-in function and access the values of the keys using the keys ad the index of the dictionary.