1. In the below elements which of them are values or an expression? eg:- values can be integer or string and expressions will be mathematical operators.

\*

'hello'

-87.8

-

/

6

Ans: ‘hello’, -87.8, 6 are the values. And expressions are

\*: multiplication,

- : subtraction,

/: division,

+: addition

2. What is the difference between string and variable?

A string is a collection of characters or alphanumeric and symbols. A string must be enclosed with single quotes or double quotes. To store the string values or any other type values we will assign with some names are called variables. A variable is name of memory location which will store a value or group of values. To naming the variables few restrictions are there, those are in the following:

* Variable name does not start with numbers and not consists white spaces
* Variable name consists only alphanumeric and \_
* To declare a variable name does not use keywords
* Variable names are case sensitive

Ex: ch=”ineuron”

In the above example, ch is a variable name and “ineuron” string value assigned to a variable name ch.

3. Describe three different data types.

Python have supporting different data types. Three of those are List, String, and Touple.

**List**: list is a collection of homogeneous or non homogeneous data type elements. We have to declare with square brackets []. It has one important feature is mutable. We can modify and edit the as how we need.

We can access the data elements using index. And it supports forward and backward data indexing.

Ex: A= [10, 3, 4, 5]

A [0] =10, A [1] =3, A [2] =4, A [3] =5 # forward indexing

A [-4] =10, A [-3] =3, A [-2] =4, A [-1] =5 # backward indexing

Forward Indexing: Index starts with 0 and ends with n-1, where n is number of elements in a list.

Backward Indexing: It starts from -n and ends with -1.

It will include starting index and excludes ending index.

Ex: A [0:2]

[10, 3], it will execute only 0 and 1 positions of the list and ignored index 2.

It supports mathematical operations while all the elements in the list are numeric.

Few of list handling functions are insert (), append (), extend (), arithmetic operations, remove (), del (), and etc.

**String:**  it is a collection of characters or alphanumeric and symbols enclosed with single or double quotes. It is immutable. It will not support mathematical operations but we can use addition operator to append string and \* operand for to repeat string multiple times.

It also supports back ward and forward indexing.

Ex: a=”ineuron”

a [0:2] =”in”

a [-3:-1] =”ro”

There are many string handling functions are available in python. Few of are, upper(), lower(), reversed(), sorted(), replace(), and etc.

**Touple:** Touple is collection of non homogeneous or homogeneous data type elements. It is immutable. And it is declared within the parenthesis ().

It also supports the backward and forward indexing.

Ex: creating touple with single element, t=(1,)

Ex: t= (1, 2, 3, 4)

t [0:2]=(1,2)

t [-4,-3]=(1,2)

We can’t perform any mathematical operations, except + and \*. + operand used for concate the tuples and \* operand used for to execute multiple times touple of elements. Only limited operations can perform on touples, few of touple handling functions are, len(), del() while all elements are numeric in touple then we can perform max(), sum() and min().

4. What is an expression made up of? What do all expressions do?

Ans: An expression is a collection of variables, operands and evaluated functions to get single value and constructed according to syntax of respective programming langouge.

It consists operands, +,-, \*, /, //, %

Variables, ex: a+b+c, where a, b, and c are variables.

And evaluated functions example, a=[1,2,3], b= sum(a)/3

Ex: a+b+c+max(a)

5. This assignment statements, like spam = 10. What is the difference between an expression and a statement?

Ans: An expression is a combination of variables, function, and operators to generate a value.

Every line in a python code is a statement, an expression also a statement.

Ex: for i in range (10):

a=i\*i # (it is an expression)

return a

In the above example all the lines are statements of python code.

6. After running the following code, what does the variable bacon contain?

bacon = 22

bacon + 1

Ans: 22

While we adding the 1 to variable bacon it displays 23 but it does not store that value it stores only 22.

7. What should the values of the following two terms be?

'spam' + 'spamspam'

'spam' \* 3

Ans: 'spam' + 'spamspam'= ’spamspamspam’

‘spam’\*3=’spamspamspam’

8. Why is eggs a valid variable name while 100 is invalid?

Variable name eggs is valid, because of variable name must be have to follow

* Variable name does not start with numbers and not consists white spaces
* Variable name can consists only alphabetic or numeric or \_ or combination.
* To declare a variable name does not use keywords
* Variable names are case sensitive

And 100 is not valid variable name because of variable name must be start with alphabetic or \_ (underscore).

9. What three functions can be used to get the integer, floating-point number, or string version of a value?

Ans: to get the integer values int() function, to get floating point numbers float(), and to get string values have to use str() or declare value with single or double quotes.

But only number values we can get in int or float format, string can store any type of values into string data type.

10. Why does this expression cause an error? How can you fix it?

'I have eaten ' + 99 + ' burritos.'

Ans: it will get an error; why because of only numerics or strings can perform addition operation. To concate these values must be declare 99 in the single quotes or double quotes as a string value or have to remove the term +99.