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.**

|  |  |
| --- | --- |
| **Expressions** | **Values** |
| \*, -, /, + | ‘hello’, -87.8, 6 |

2. What is the difference between string and variable?

**Ans.**

**Variables** are used to store value of any data type. Rules for defining variable name is predefined.

**String** is a data type which comes under sequence type classification of python data type. It is a group of one or more characters which can be represented in ‘ ’, “ ”, ””” “”” and variables can be used to store string.

3. Describe three different data types.

**Integer:**

* Integer comes under numeric data type which represents numeric value
* It can take any positive or negative value including zero (whole numbers) excluding fraction or decimal values.
* It can be represented by <class ‘int’>
* Theoretically, it is unbounded.
* Ex. 1, 0, 1444, -20, …

**List:**

* It is in-build data type in python which comes under sequence data type classification.
* It can store multiple values of same or different data types.
* List can be created by giving values inside the [ ] square bracket.
* Ex. List = [1, 2,” name”, 2.4, True, 4+j2]

**Tuple:**

* It’s also in-build data type in python which comes under sequence data type classification.
* Tuple and List are same but the only difference between them is that Tuples are **immutable** means it cannot be changed once initialised.
* Tuple can be created by giving values inside the ( ) round bracket.
* Ex. Tuple = (1, 2,” name”, 2.4, False, 4+j2)

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

**Ans.** Expression consists of variables(identifiers), operators, values(literals). Expressions evaluate complex arithmetic, relational, logical, bitwise operations.

Ex. 2 > 3, 2 + 3, a > 2

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

**Ans.** Statement is nothing but a line of code that is going to be executed by interpreter. Here, spam = 10, “10” is the expression and “spam = 10” is the assignment statement which assign value 10 to variable spam.

Expressions is going to be evaluated and produces a value and statements do not produce any value. Statement gets only executed. In assignment statement, expressions are on the right side of assignment operator.

Ex. Expression: 5+2\*1-(1+100)

Statement: value = 5+2\*1-(1+100)

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

bacon = 22

bacon + 1

**Ans.** bacon contain **22.**

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

'spam' + 'spamspam'

'spam' \* 3

**Ans.** 'spam' + 'spamspam' 🡪 ‘spamspamspam’ .... (concatenation)

'spam' \* 3 🡪 ‘spamspamspam’

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

**Ans.**

In python, there are some predefined rules for defining variable name.

Rule: Variable name can start with a letter or the underscore character.

“eggs” is valid variable name according to this rule.

Rule: Variable name cannot start with a number.

“100” in invalid variable name according to this rule.

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

**Ans.**  **int(), float(), str()** functions can be used to get the integer, floating-point number or string version of value respectively.

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

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

**Ans.** if ‘+’ occurs between strings, concatenation of strings takes place. So, the operands should be of **same data type as string** otherwise it will give an error.

As 99 is of integer datatype. It can’t be concatenated. We have to convert it in to string.

**Fixed Expression:** 'I have eaten ' + **'** 99 **'** + ' burritos.'