**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:-** In above elements values are:- ‘hello’, -87.8, 6

And the expressions are :- \* , -, /, + these all are mathematical expressions.

**2. What is the difference between string and variable?**

**Ans:-** String is a data type while Variables are the symbol or text in which we store any kind of data.

**3. Describe three different data types.**

**Ans:-** There are different built in data types in Python. Some of them are follow.

* List :- List are used to store collection of data , the data stored between square brackets in list. Lists are ordered means items enclosed in it have a defined order, and that order will not change. It is mutable data type, and it can also be accept duplicates in it.

e.g- List1 = [“apple”, “mango”, “grapes”, “banana”]

* Tuples:- Tuples are used to store multiple items in a single variable. Tuples are also ordered. This is a immutable data type, which means that you can not change tuples after defining it. It allows duplicates. Tuples are written in round brackets.

e.g- tuple1= (“apple”, “mango”, “grapes”, “banana”)

* Sets:- It is also used to store multiple items within single variable. This is unordered data type, means that the items in set do not have defined order.

Sets are immutable, we cannot change sets once we create it. But we can add and remove items from it. Duplicates are not allowed in sets.

Sets are written in curly brackets.

e.g- {“apple”, “mango”, “grapes”, “banana”}

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

**Ans:-** Expression is a combination of values, variables, operators which produce a new value after evaluating it. It can be simple as a single value or may be complex as a combination of multiple sub-expressions.

In python or any programming language expressions are the fundamental building blocks that are evaluate to produce new results. All expressions have value which can be any data type.

e.g- c=a + b this is a simple example expression which contains two values and one operator sign to result a new value.

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

**Ans:-** *Expression-*

* It is a combination of values, operators, variables that evaluate to generate single value.
* Examples of expressions: 2 + 3, x \* 5, len(my\_list), "Hello" + "World".
* Expressions are used within a statement.

*Statements-*

* Statement is a block of code that performs certain actions.
* This contains multiple expressions in it.
* Statements do not evaluate value.
* Examples of statements- ‘if’ statement, ‘for’ statement, ‘with’ statement etc.

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

bacon = 22

bacon + 1

**Ans:-** After evaluatingabove expression result will be (**bacon = 23**)

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

'spam' + 'spamspam' **(result= spamspamspam)**

'spam' \* 3 **(result= spamspamspam)**

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

**Ans:-** We cannot use number as a variable name instead of we can use the combination of text and number as a variable name. By using combination of egg and 100 we can create multiple variable names.

For e.g eggs100, eggs\_100 etc.

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

**Ans:-** To get integer value – x= int(21)

To get float value – y= float(2.33)

To get string value – z= str(“Python”)

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

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

**Ans:-** In above expression we are trying to concatenate numeric value with string. In python we cannot concatenate numeric values. Instead we can write this number in quotient (it will became string) and can able to concatenate all words.

Corrected code:- ‘I have eaten ‘ + ‘99’ + ‘burritos.’