Solutions:

**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.**

\* (expression)

'hello' ( value)

-87.8 (value)

- (expression)

/ (expression)

* (expression)

6 (value)

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

🡪 Simply string is a datatype and variable is something that stores any data in it.  string data type is primarily used for reading and writing data. Variables are references that we create to refer to other values in our program later. String deals with character type data where a variable can contain any kind of datatype in it.

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

🡪

1. **Number:**  It contains all the numerical data. It is further divided into 3 types for easy mathematical operation.

* int – holds signed integers of non-limited length.
* float- holds floating precision numbers and it’s accurate up to 15 decimal places.
* complex- holds complex numbers.

1. **String:** The string is a sequence of characters. Python supports Unicode characters. Generally, strings are represented by either single or double-quotes.
2. **List**:  list is an ordered sequence of some data written using square brackets([ ]) and commas(,). It can simultaneously hold different types of data.

Ex: b=[“x”, ”y”, 1, 2 ] ; here b is a list which contains both string and integer in it.

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

🡪  An expression is a combination of operators, constants and variables. An expression may consist of one or more operands, and zero or more operators to produce a value. Expressions represent something, like a number, a string, or an instance of a class. Any value is an expression.

 If you type an expression on the command line, the interpreter evaluates it and displays the result.

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

🡪 **Expression** is Anything you can legally say in a spot where a *value* is required. Typically composed of *literals, variables, operators, functions,* and *subroutine* calls, not necessarily in that order.

Whereas **statement** is a *command* to the computer about what to do next, like a step in a recipe: "Add marmalade to batter and mix until mixed." A statement is distinguished from a *declaration*, which doesn't tell the computer to do anything, but just to learn something.

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

bacon = 22

bacon + 1

* 23

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

'spam' + 'spamspam' 🡪 'spamspamspam'

'spam' \* 3 🡪 'spamspamspam'

They are same.

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

**🡪** variable name cant be a numeric type data. Rather it can be a character type data. That’s why.

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

**🡪** int(), float(), str()

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

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

* Because the 99 in the middle is a integer while the other parts are strings. Concatenation must be between same datatypes. The correct expression will be:

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