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

Answer: In the given list, the following are values or expressions:

Values: ‘hello’, -87.8 and 6

Expression: \*,-,/,+

2. What is the difference between string and variable?

Answer: String:

A string is a data type that represents a sequence of characters. It can include letters, numbers, symbols, and whitespace.

Strings are often used to store and manipulate text data.

In many programming languages, strings are typically enclosed in single (' '), double (" "), or triple (''' ''' or """ """) quotation marks.

Examples of strings: "Hello, World!", '123', "Python is fun."

Variable: A variable is a symbolic name or identifier associated with a storage location in a program.

It is used to store and represent data values or objects.

The data type of a variable defines the type of values it can hold, such as integers, floats, strings, etc.

Variables allow for the storage and manipulation of values within a program.

3. Describe three different data types.

Answer: Integer (int):

An integer is a whole number without any decimal point.

Example: x = 5

Floating-point (float):

A floating-point number is a number that has a decimal point or is expressed in exponential notation.

Example: y = 3.14

String (str):

A string is a sequence of characters, enclosed within single or double quotes.

Example: message = "Hello, World!"

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

Answer: Expression:

An expression in Python is a combination of values, variables, operators, and function calls that results in a single value. It can be as simple as a single variable or a complex combination of various elements. Expressions are the building blocks of Python programs and are evaluated to produce a value.

Components of an Expression:

Values: Literal values like numbers or strings.

Variables: Names that represent values stored in memory.

Operators: Symbols that perform operations on values and variables.

Function Calls: Invocations of functions that return a value.

What Expressions Do in Python:

Evaluation: Expressions are evaluated to produce a single value. The result can be a number, string, boolean, or any other data type.

Side Effects: Some expressions may have side effects, such as modifying variables or causing I/O operations. However, pure expressions, which only depend on their inputs and don't cause side effects, are common in Python.

Assignment: Expressions can be used to assign values to variables.

Control Flow: Expressions are often used within control flow statements (e.g., if, while) to make decisions based on their evaluation.

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

Answere: An assignment statement, like spam = 10, combines an expression (10) with the action of assigning it to a variable (spam).

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

bacon = 22

bacon + 1

Answer: 23

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

'spam' + 'spamspam'

'spam' \* 3

Answer: 'spamspamspam'

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

Answer: eggs is a valid variable name, but 100 is not valid according to the rules for Python variable names. If you want to use a number in a variable name, it should come after a letter or an underscore. For example, eggs100 would be a valid variable name.

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

Answer: In Python, you can use the following three functions to convert a value to an integer, a floating-point number, or a string:

Integer Conversion:

Function: int()

Example value = 5.6

integer\_value = int(value)

Function: float()

Example: value = 10

float\_value = float(value)

String Conversion:

Function: str()

Example: value = 42

string\_value = str(value)

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

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

Answer: The expression 'I have eaten ' + 99 + ' burritos.' causes an error because you are trying to concatenate a string with an integer without converting the integer to a string explicitly. In Python, you cannot concatenate different types without converting them to a common type first.