**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.**

**Values:**

'hello' is a string value.

-87.8 is a floating-point value.

6 is an integer value.

**Expressions:**

- is a subtraction operator.

/ is a division operator.

+ is an addition operator.

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

**A string** is a sequence of characters. It is a data type that represents text. Strings are enclosed in quotation marks in most programming languages. For example, the string "hello" is a sequence of five characters: "h", "e", "l", "l", and "o".

**A variable** is a named location in memory that stores data. It is a data type that can store any type of data, including strings, numbers, and Boolean values. Variables are declared with a name and a data type. For example, the variable my string could be declared as a string.

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

Below are three different data types:

**Integer:** An integer is a whole number, positive or negative. Integers can be represented in several different ways, such as decimal, hexadecimal, and octal. For example, the integer 123 is a decimal integer, while the integer 0xF is a hexadecimal integer.

**Floating-point:** A floating-point number is a number that can represent both whole and fractional parts. Floating-point numbers are typically represented in scientific notation, such as 1.234e+03, which represents the number 1234.

**String:** A string is a sequence of characters. Strings are enclosed in quotation marks in most programming languages. For example, the string "hello" is a sequence of five characters: "h", "e", "l", "l", and "o".

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

An expression is a combination of variables, operators, and values that evaluates to a single value. Expressions can be used to perform a variety of operations, such as arithmetic, logical, and comparison operations.The basic building blocks of an expression are:

**Variables**: Variables are named locations in memory that store data.

**Operators:** Operators are symbols that perform operations on data.

**Values:** Values are the data that is operated on by the operators.

Expressions can be made up of any combination of variables, operators, and values. For example, the expression 2 + 3 \* 4 is made up of the variables 2 and 3, the operator +, and the value 4. This expression evaluates to the value 14, which is the result of adding 2 to the product of 3 and 4.

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

An expression is a combination of variables, operators, and values that evaluates to a single value. An assignment statement is a statement that assigns a value to a variable.

The main difference between an expression and a statement is that an expression evaluates to a value, while a statement does not. For example, the expression 2 + 3 \* 4 evaluates to the value 14, while the statement spam = 10 does not evaluate to a value.

The assignment statement spam = 10 is an example of a statement that assigns a value to a variable. The variable spam is assigned the value 10. The value of spam can then be used in other expressions or statements.

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

**bacon = 22**

**bacon + 1**

The variable bacon will contain the value 22 after running the code. The first line of code assigns the value 22 to the variable bacon. The second line of code is an expression that adds 1 to the value of bacon. However, expressions do not store their results, so the value of bacon is not changed. Therefore, after running the code, the variable bacon will still contain the value 22.

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

**'spam' + 'spamspam'**

**'spam' \* 3**

The values of the following two terms should be:

**The first term**, 'spam' + 'spamspam', is the concatenation of the two strings 'spam' and 'spamspam'. The concatenation operator (+) takes two strings and returns a new string that is the concatenation of the two strings. In this case, the concatenation of 'spam' and 'spamspam' is 'spamspamspam'.

**The second term**, 'spam' \* 3, is the repetition of the string 'spam' three times. The multiplication operator (\*) takes a string and an integer and returns a new string that is the repetition of the string the specified number of times. In this case, the repetition of 'spam' three times is 'spamspamspam'.

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

In Python, variable names can be made up of letters, numbers, and underscores. However, variable names cannot start with a number. Additionally, variable names cannot be the same as keywords in Python.

The variable name eggs follows these rules, so it is a valid variable name. The variable name 100 starts with a number, so it is an invalid variable name.

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

Below are the three functions that can be used to get the integer, floating-point number, or string version of a value:

**int():** The int() function converts a value to an integer. For example, the expression int("123") converts the string "123" to the integer 123.

**float():** The float() function converts a value to a floating-point number. For example, the expression float("1.23") converts the string "1.23" to the floating-point number 1.23.

**str():** The str() function converts a value to a string. For example, the expression str(123) converts the integer 123 to the string "123".

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

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

The expression 'I have eaten ' + 99 + ' burritos.' causes an error because the + operator cannot be used to concatenate a string and an integer. The + operator can only be used to concatenate two strings or two integers.

To fix the error, you can use the str() function to convert the integer 99 to a string.