1. Among the elements listed:

\* is an expression (multiplication operator).

'hello' is a value (string).

-87.8 is a value (floating-point number).

- is an expression (subtraction or negation operator).

/ is an expression (division operator).

+ is an expression (addition operator).

6 is a value (integer).

2. Difference between string and variable:

A string is a type of data that consists of a sequence of characters. It can be any combination of letters, numbers, symbols, and spaces enclosed within quotation marks. For example, "hello world".

A variable, on the other hand, is a name given to a storage area that your programs can manipulate. The variable can contain data like strings, numbers, etc. For instance, name = "John"; here, name is a variable that holds the string value "John".

3. Three different data types:

Integer: Represents whole numbers, e.g., -1, 0, 5, 10.

Float: Represents numbers with decimal points, e.g., 3.14, -0.001.

String: Represents sequences of characters, e.g., "hello", 'world'.

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

An expression is a combination of values, variables, and operators. All expressions evaluate to a single value. For example, the expression 2 + 3 evaluates to 5.

5. Difference between an expression and a statement:

An expression evaluates to a single value, while a statement performs an action. The assignment spam = 10 is a statement. The right-hand side (10) is an expression that evaluates to the value 10, which the statement then assigns to the variable spam.

6. After running the code:

bacon will still contain the value 22. This is because the statement bacon + 1 computes the value 23 but doesn't store it anywhere.

7. Values of the following two terms:

'spam' + 'spamspam' evaluates to 'spamspamspam'.

'spam' \* 3 also evaluates to 'spamspamspam'.

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

Variable names in most programming languages, including Python, must start with a letter (a-z, A-Z) or an underscore \_. They cannot start with a number. Thus, eggs is a valid variable name, while 100 is not.

9. Three functions to get the integer, floating-point number, or string version of a value:

int(): Converts a value to an integer, e.g., int(3.5) results in 3.

float(): Converts a value to a floating-point number, e.g., float("3.14") results in 3.14.

str(): Converts a value to a string, e.g., str(100) results in '100'.

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

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

This expression causes an error because Python does not allow concatenation of strings with integers directly. To fix it, you need to convert the integer 99 to a string using the str() function:

Fixed Expression: 'I have eaten ' + str(99) + ' burritos.'