**1. Values vs Expressions**

* **Values**: These are data that represent something, such as an integer, string, or boolean.
  + **Examples of values**:
    - 'hello' (string value)
    - -87.8 (float value)
    - 6 (integer value)
* **Expressions**: These are combinations of values, variables, and operators that can be evaluated to produce a value.
  + **Examples of expressions**:
    - - (unary minus operator)
    - / (division operator)
    - + (addition operator)

So, in your list:

* 'hello', -87.8, and 6 are **values**.
* -, /, and + are **expressions**.

**2. Difference between String and Variable**

* **String**: A string is a data type used to represent text. It is enclosed in quotation marks (either single ' or double ").  
  Example: 'hello' is a string.
* **Variable**: A variable is a name that is used to store a value (could be of any data type, including strings, numbers, etc.).  
  Example: name = 'John'—name is the variable storing the string 'John'.

**3. Three Different Data Types**

* **Integer (int)**: A whole number without a decimal point.  
  Example: 5, -42
* **Float (float)**: A number that has a decimal point.  
  Example: 3.14, -0.99
* **String (str)**: A sequence of characters enclosed in quotation marks.  
  Example: 'hello', '123'

**4. What is an Expression Made Up of? What Do All Expressions Do?**

An **expression** is made up of:

* **Values**: like numbers, strings, etc.
* **Operators**: like +, -, \*, /, etc.
* **Variables**: names that hold values.
* **Function calls**: like len(), max(), etc.

All **expressions** are evaluated to produce a value. For example, 2 + 3 is an expression that evaluates to 5.

**5. Difference Between an Expression and a Statement**

* **Expression**: An expression is a piece of code that evaluates to a value. For example, 2 + 2 is an expression.
* **Statement**: A statement is a line of code that performs an action but does not return a value. For example, spam = 10 is a statement, as it assigns a value to the variable spam but doesn't "evaluate" to something.

**6. What Does the Variable bacon Contain?**

python

Copy

bacon = 22

bacon + 1

* The variable bacon will contain 22.
* The statement bacon + 1 does not affect bacon because it is an expression and is not assigned back to bacon. You would need to write bacon = bacon + 1 or bacon += 1 to update bacon's value.

**7. Values of the Two Terms**

python

Copy

'spam' + 'spamspam' # Concatenation

'spam' \* 3 # Repetition

* 'spam' + 'spamspam' results in 'spamspamspam' (string concatenation).
* 'spam' \* 3 results in 'spamspamspam' (string repetition).

**8. Why is eggs a Valid Variable Name While 100 is Invalid?**

* **eggs** is a valid variable name because it starts with a letter and is followed by letters or numbers (valid characters for a variable name).
* **100** is invalid because variable names cannot begin with a number. Variable names must begin with a letter or an underscore.

**9. Three Functions for Integer, Floating-Point, or String Version of a Value**

* **int()**: Converts a value to an integer.  
  Example: int('5') results in 5.
* **float()**: Converts a value to a floating-point number.  
  Example: float('3.14') results in 3.14.
* **str()**: Converts a value to a string.  
  Example: str(10) results in '10'.

**10. Why Does This Expression Cause an Error? How to Fix It?**

python

Copy

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

This causes an error because you are trying to concatenate a string ('I have eaten ') with an integer (99). Python does not allow concatenating strings and integers directly.

**Fix**: Convert the integer 99 to a string using str():

python

Copy

'I have eaten ' + str(99) + ' burritos.'

This will correctly output: 'I have eaten 99 burritos.'