**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?**

A string is a sequence of characters. They are immutable. Ex: a= ”Aakash”. Here ‘a’ is of string type.   
A variable an empty container in which we can input any value. A variable can be of any type ex. Int, float, list, tuple etc.

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

**Integer:** This data type contains only the integer value. There is no limit, an integer can be of any size(it is constrained by the amount of memory our system has).

>>type(10)

Output: <class ‘int’>

**Float**: The float type designates a floating-point number. Float values are values with a decimal point. Ex. a = 4.9

>>type(a)

Output: <class ‘float’>

**Complex Numbers:** Complex numbers are specified as (real part)+(imaginary part)j. Here only j is used with the imaginary part, if any other character is used then Python will throw an error. Ex. a= 5+9j

>>type(a)

Output: <class ‘complex’>

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

An **expression** is a combination of values, variables, operators, and calls to functions. Expressions are representations of value. For example, any string is also an expression since it represents the value of the string as well. All expressions are needed to be evaluated. They perform a task. For ex. a+2. In this expression, 2 is being added to the variable a.

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

**Expressions** are representations of value. They are different from statement in the fact that statements do something while expressions are representation of value.

For Ex. 2+3

A **statement** is a unit of code that has an effect, like creating a variable or displaying a value.

For ex: n=17

print(n)

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

**bacon = 22**

**bacon + 1**

After running the code, the variable name bacon contains 22 value which is its initial value. As in the second line bacon+1, we do not assign it back to bacon variable. So, the final value will be 22 only.

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

**'spam' + 'spamspam'** 🡪 spamspamspam

**'spam' \* 3** 🡪 spamspamspam

Both values are same in output.

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

**Rules:**

1. A variable name can be of any length. From a single character name( like a or b) to a long descriptive name (like number\_of\_students).
2. A variable name must start with a letter or the underscore.
3. **A variable name can only contain alpha-numeric characters and underscores (A-z, 0–9, and \_ )**
4. Variable names in Python are case-sensitive(Eggs, eggs, EgGs are all different variable names)
5. **A variable name cannot start with a number(like 100- it is not permissible)**
6. A variable name should not be among Python reserved keywords. Each keyword has a special meaning and cannot be used as a variable name.

As per the above rules, eggs is a valid name and 100 is an invalid variable name.  
  
I have written a medium post over this: <https://medium.com/@aakashgoyal25193/variable-names-in-python-9a3646eeee9b>

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

**Int(), float(), str()**: These three functions are used to get the integer, floating-point number, or string version of a value.

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

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

Here we are trying to concatenate integer value with string value which in not allowed.   
We can simply fix this by type casting 99 into string.  
'I have eaten ' + '99' + ' burritos.'

Output: 'I have eaten 99 burritos.'