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.

**Ans:** Values : 'hello' , -87.8, 6

Expression : \* - / +

2. What is the difference between string and variable?

**Ans**. **A string** is a data type used to represent text enclosed in single or double quotes.

Eg. Name= ‘tasnim’

A **variable** is a name that refers to a value stored in memory.

It can hold different types of data, including strings, numbers, lists, dictionaries, etc.

Eg. X=90

3. Describe three different data types.

**Ans**. **Numeric Types:**

* + **int**: Integer numbers (e.g., 5, -3, 100).
  + **float**: Floating-point numbers, also known as decimal numbers (e.g., 3.14, -0.5, 2.0).

**Sequence Types:**

* + **str**: String, a sequence of characters (e.g., "hello", 'python', "123").
  + **list**: Ordered collection of items (e.g., [1, 2, 3], ['a', 'b', 'c'], [1, 'hello', 3.14]).
  + **tuple**: Immutable ordered collection of items (e.g., (1, 2, 3), ('a', 'b', 'c'), (1, 'hello', 3.14)).

**Boolean Type:**

* + **bool**: Boolean value representing True or False.

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

Ans. An expression in python is made up of the following:

**Operand:** This is a data value or a variable used in the expression.

**Eg.** **5**, **"hello"**,

**Operator:** This is a symbol that represents a mathematical, logical, or bitwise operation to be performed on the operands. Eg **+**, **-**, **\***, **/** for arithmetic operations, **==**, **!=**, **<**, **>** for comparison operations, **and**, **or**, **not** for logical operations

**Function Calls:** A function call can also be considered an expression if it returns a value. Eg: **len("hello")** is an expression that returns the length of the string **"hello"**.

**Subexpressions:** Expressions can contain other expressions within them. For example, **(x + y)** is a subexpression within the expression **2 \* (x + y)**.

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

|  |  |
| --- | --- |
| Expression | Statement |
| An expression solves for a single value | Usually multiple lines of codes |
| Returns a value | Not necessary to return a value |
| Uses functions as well. | Executed sequentially by Python's interpreter. |

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

bacon = 22

bacon + 1

**Ans.** bacon+1 prints 23 where as bacon contains the value=22

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

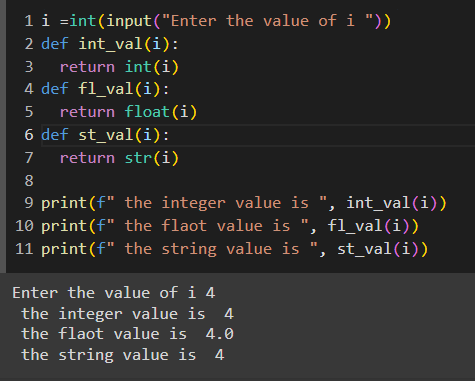
'spam' + 'spamspam' –‘spamspamspam’

'spam' \* 3 – ‘spamspamspam’

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

**Ans**. A number cannot be used as a variable name. Hence 100 is invalid.

9. What three functions can be 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.'

**Ans.** We cannot add a string and a number