1. In the below elements which of them are values or an expression? e.g.: - values can be

integer or string and expressions will be mathematical operators.

\*

‘hello’

-87.8

-

/

+

6

**Answer**: - values can be an integer and string so in given option there is three values I can see.

**> ’hello’**

**> -87.8**

**> 6**

Expression will be a mathematical operator so there are three expressions

* **-**
* **/**
* **+**

2. What is the difference between string and variable?

**Answer**- **‘string’**

**>string** is a data type

>string is a sequence of character enclosed in quotes.

Primarily we write strings single (‘’) or doble quote (“”).

>strings are immutable. If I want to modify it then I have to create another string

>in string we can write letters, numbers, and other character

**Variable=**

> e.g. A=10

>A is the variable works as a container who holds the value of 10.

>variable is representing memory locations where data is stored

>variable contain alphabet digits and underscores

>variable also holds the strings, integers and more other data types

>with variable we can manipulate data

>we can assign string to a variable

3. Describe three different data types.

Answer-data types are used to define a variable so there are some primarily used data types:

**String**: string is a sequence of characters enclosed in quotes. E.g., "hi","10",""@".

>purpose of strings is to store human-readable text, like words and sentences.

**[list]:** Lists are ordered, mutable collections of elements, enclosed in square brackets.

>Lists are used to store and manage multiple related items in a single data.

**set**: Sets are unordered collections of unique elements, enclosed in curly braces ({}).

>Sets support various set operations like union, intersection, difference, and symmetric difference, which are useful in mathematical and logical operations.

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

**Answer**: An expression in programming is made up of a combination of components like variable, parenthesis, square brackets, curly braces, operators

>expression is used for many purposes in programming like manipulate data, calculation, logical evaluation and it is helpful for performing a particular task

5. This assignment statements, like spam = 10. What is the difference between an

expression and a statement?

**Answer:**

|  |  |
| --- | --- |
| expression | statement |
| >expression always return a value  >function is also an expression  >its primary purpose to compute a result value  >expression can be used as a part of larger statement  >e.g.: X+2(an expression)  2\*2(also an expression) | >statement is a programming instruction that does something.  >a statement never returns a value  >cannot print any result  >e.g.: x=1  Y=x+1  Print(y) |

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

bacon = 22

bacon + 1

**Answer:**  if we **bacon = 22**

**bacon + 1** run this code as it is so the output will be 22.

So, if we want to update the value of bacon, we should use an assignment statement

**Bacon=22**

**Bacon=bacon+1**

**Print(bacon)**

So, the output is **23.**

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

‘Spam’ + ‘spamspam’

‘spam’ \* 3

**Answer**:

*‘Spam’\*3* **🡨** if we print this value so the output will be ‘*spamspamspam’.*

* It multiplies spam three times

*‘Spam’ + ‘spamspam’* **<-** the output will be *‘spamspamspam’*

* This term performs string concatenation

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

**Answer:**

variable name cannot start with digit (0-9). And also, cannot be the same as python keywords.'100' violates the rules by starting with a digit and is thus an invalid variable name.

9. What three functions can be used to get the integer, floating-point number, or string

version of a value?

**Answer:**

To get the integer representation we can use the following function

*int () #*convert value to an integer

*floor () #*used to round a floating-point number down to the nearest integer.

*round () #*used to round a floating-point number to the nearest integer

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

‘I have eaten’ + 99 + ‘burritos’

Answer:

This expression shows the error because it attempts to concatenate a string with an integer which is not allowed in python.

To fix the error we need to covert integer to a string.

Food=’ I have eaten’ + str (99) + ‘burritos’

Print (Food)

#and the output will be ‘I have eaten 99 burritos’