# **Assignment\_1**

# 

### 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.

### \*

### ‘hello’

### -87.8

### -

### /

### +

### 6

### Answer:

| **Values (Arithmetic Operators)** | **Expression (Integer, Float & String)** |
| --- | --- |
| \* | ‘hello’ |
| - | -87.8 |
| / | 6 |
| + |  |

### 2. What is the difference between string and variable?

Answer:

String:

A string is a data type. It can be used to store inputs from the user. It is also used to store data in the form of alphabets, digits, symbols, and many more.

E.g.: ‘Sarvesh’, ‘variable’

Variable:

A variable is used to store the data type. It can store integers, float, string, complex, boolean, etc.

E.g.: name = ‘Sarvesh’ (Here name is a variable that stores string ‘Sarvesh’).

### 3. Describe three different data types.

Answer:

Data types can be explained as different types of data. It stores this data for various operations.

The data types in Python are:

1. Integer
2. Character/ String
3. Boolean
4. Float
5. Complex

Integer:

It stores digits as data type in variables. These integers can be used for various mathematical calculations.

E.g.: num1 = 25

(Here 25 is stored into num1 as integer)

String:

A string stores characters, numbers and symbols. These strings are highly readable by the user, and hence the input from the user is taken as string by default. It is usually stored in between single quotes (‘ ‘) or double quotes (“ “).

E.g.: address: ‘Pune, Maharashtra’

(Here ‘Pune, Maharashtra’ is a string. It is stored in the variable address)

Boolean:

Boolean is a data type that only has two values. It is stored as **True** or **False**. This data type is used to indicate the state of the statement.

E.g.: Boolean can be used to store whether the fan switch is ON or OFF. ON can be stated as **True** and OFF can be stated as **False**.

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

An expression consists of operators and operands. There can be multiple operators. And it is necessary to have at least two operands.

Operators and symbols that signify the expression that needs to be performed.

E.g.: \*, /, +, -, etc.

An operand is a number or a variable on which the expression is performed.

E.g.: 2, x, etc.

Both operand and operator come together to form an expression. For instance, in the expression given below:

5 + 6 = 11.

Here ‘5’ and ‘6’ are operand whereas ‘+’ sign is operator.

Expressions are used to solve any mathematical problem. For instance they can be used to append the variable

temp = temp + 1

Here it is stated that add 1 to the temp variable and store it into the same variable.

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

Answer:

A statement is used to state or assign a value to the variable. Here (spam = 10), spam is a variable and integer 10 is assigned it.

An expression is a mathematical operation that is solved.

Consider an example:

X = 4 - 2

This is an expression as it has two operands (4 and 2) and an operator (-).

This expression is solved.

X = 2

Now this derived line is a statement. It assigns the value of X as 2.

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

### bacon = 22

### bacon + 1

Answer:

The value of bacon will still be 22.

This happens because line 2 is an expression and hence it doesn’t change the value of the variable.

If the value of the variable is supposed to be changed, then the 2nd line can be writer as below:

bacon = bacon + 1

This will append the value of bacon by 1, hence making the final value of it as 23.

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

### 'spam' + 'spamspam'

### 'spam' \* 3

Answer:

The value of first line ('spam' + 'spamspam'):  
‘spamspamspam’

Here two strings ‘spam’ and ‘spamspam’ are concatenated together.

The value of second line (‘spam’ \* 3):

‘spamspamspam’

Here the string ‘spam’ is repeated 3 times.

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

Answer:

There are certain rules when it comes to naming a variable.

1. All uppercase and lowercase letters are allowed.
2. Only one symbol i.e. underscore (\_) can be used to name a variable.
3. The variable can contain digits from 0 to 9.
4. The first character of a variable cannot be a number.

Now consider the examples:

**eggs** can be used as a variable since it satisfies all the above conditions.

**100** can not be used as a variable since it doesn’t satisfy condition 4 (The first character of a variable cannot be a number).

### 

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

Answer:

The process of changing the version of a variable is called typecasting.

It can be done by using the function of the data it has to be type casted into.

Value to integer:

The int() function is used here.

E.g.: var = ‘157’

var = int(var)

This will convert string to an integer

Value to float:

The float() function is used here.

E.g.: var = ‘157.58’

var = float(var)

This will convert string to a float.

Value to string:

The int() function is used here.

E.g.: var = 157

var = str(var)

This will convert integer to an string

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

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

It seems the user wants to print 3 values at a time. He/ she wants the output to be:

‘I have eaten 99 burritos.’

But the syntax for the same is incorrect.

The above output can be printed by:

print(‘I have eaten ’)

print(99)

print(‘ burritos’)

Here all three values are printed separately.

OR

print(‘I have eaten’, 99,’ burritos.’)

Here all three values are printed together. Only (+) in the questions needs to be replaced by (,).

### 