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

-

/

**Answer ---**

\* : expression

'hello' :value

-87.8 :value

- :expression

/ :expression

6 :value

2. What is the difference between string and variable?

**Answer---**

A Variable is a store of information, and a String is a type of information you would store in a Variable. A String is usually words, enclosed with ""

Eg -- String x ="Welcome to Python" X is the Variable, and we declared it as a String, and used the single = to assign the text to it .

3. Describe three different data types.

**Answer---**

. Data types are the classification or categorization of data items. It represents the kind of value that tells what operations can be performed on a particular data. Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

Variables can hold values, and every value has a data-type. Python is a dynamically typed language; hence we do not need to define the type of the variable while declaring it. The interpreter implicitly binds the value with its type.

Example a = 5

The variable **a** holds integer value five and we did not define its type. Python interpreter will automatically interpret variables **a** as an integer type.

A variable can hold different types of values . For example, a person's name must be stored as a string whereas its id must be stored as an integer.

Python provides various standard data types that define the storage method on each of them. The data types defined in Python are given below.

1. [Numbers](https://www.javatpoint.com/python-data-types#numbers)
2. [Sequence Type](https://www.javatpoint.com/python-data-types#SequenceType)
3. [Boolean](https://www.javatpoint.com/python-data-types#Boolean)
4. [Set](https://www.javatpoint.com/python-data-types#Set)
5. [Dictionary](https://www.javatpoint.com/python-data-types#dictionary)



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

**Answer—**

A combination of operands and operators is called an **expression**. The expression in Python produces some value or result after being interpreted by the Python interpreter. An expression in Python is a combination of operators and operands.

An example of expression can be : x = x + 10*x*=*x*+10. In this expression, the first 1010 is added to the variable x. After the addition is performed, the result is assigned to the variable x.

**Example :**

x = 25 *# a statement*

x = x + 10 *# an expression*

An expression in Python is very different from statements in Python. A statement is not evaluated for some results. A statement is used for creating variables or for displaying values.

The **expression** in Python can be considered as a logical line of code that is evaluated to obtain some result. If there are various operators in an expression then the operators are resolved based on their precedence

An expression in Python can contain **identifiers**, **operators**, and **operands**. Let us briefly discuss them.

An **identifier** is a name that is used to define and identify a class, variable, or function in Python.

An **operand** is an object that is operated on. On the other hand, an **operator** is a special symbol that performs the arithmetic or logical computations on the operands.

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

**Answer----**

| **Statement in Python** | **Expression in Python** |
| --- | --- |
| A statement in Python is used for creating variables or for displaying values. | The expression in Python produces some value or result after being interpreted by the Python interpreter. |
| A statement in Python is not evaluated for some results. | An expression in Python is evaluated for some results. |
| The execution of a statement changes the state of the variable. | The expression evaluation does not result in any state change. |
| A statement can be an expression. | An expression is not a statement. |
| **Example :** x = 3*x*=3. **Output :** 33 | **Example**: x = 3 + 6*x*=3+6. **Output :** 99 |

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

bacon = 22

bacon + 1

**Answer---**

The bacon variable will be set to 20. The bacon + 1 expression does not reassigning the value in bacon (that would need an assignment statement like this: bacon = bacon + 1)

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

'spam' + 'spamspam'

'spam' \* 3

**Answer----**

Both expressions will evaluate to the string 'spamspamspam'

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

Because Variable names cannot begin with a number

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

Answer—

The int(), float(), and str() functions will evaluate to the integer, floating-point number, and string versions of the value passed to them.

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

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

Answer—

The expression causes an error because 99 is an integer, and only strings can be concatenated to other strings with the + operator. The correct way is ‘I have eaten ' + str(99) + ' burritos.'.