

# Assignment 1

**Q1: 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.**

Answer: -

- \* Expression
- 'hello' value
- 87.8 value
- expression
- / Expression
- + expression
- 6 value

**Q2: 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 " "(Double quote)

**Variable:** - A variable is basically the name of the location in the primary memory of our computer. This can be chosen by us.

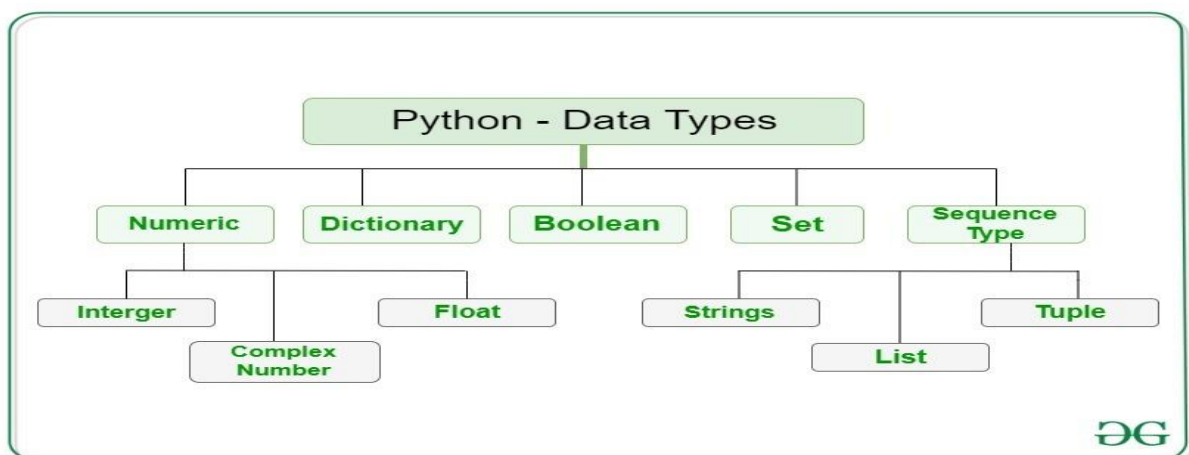
**String:** - A string is the values inside the quotes assigned to a string.

Example = Variable is `my_var = 3`

example = String is "my name is terry"

E.g. String `x = "Welcome to SoloLearn"` X is the Variable, and we declared it as a String, use the single = to assign the text to it.

**Q3: Describe three different data types.**



**Numeric Data Type:** In Python, numeric data type represents the data which has numeric value.

Integer, Float, Complex numbers are the numeric data type.

**Sequence Data Type:** - In Python, sequence is the ordered collection of similar or different data types. Sequences allows to store multiple values in an organized and efficient fashion.

- 1) **String:** - A string is a collection of one or more characters put in a single quote, double-quote or triple quote. In python there is no character data type, a character is a string of length one.
- 2) **List:** -List are just like the arrays, declared in other languages which is a ordered collection of data. It is very flexible as the items in a list do not need to be of the same type.  
Lists in Python can be created by just placing the sequence inside the square brackets [].

Initial blank List:

```
[ ]
```

List with the use of String:

```
['GeeksForGeeks']
```

Multi-Dimensional List:

```
[['Geeks', 'For'], ['Geeks']]
```

- 3) **Tuple:** - Just like list, tuple is also an ordered collection of Python objects. The only difference between tuple and list is that tuples are immutable i.e., tuples cannot be modified after it is created.

Initial empty Tuple:

```
( )
```

Tuple with the use of String:

```
('Geeks', 'For')
```

Tuple using List:

```
(1, 2, 4, 5, 6)
```

Tuple with the use of function:

```
('G', 'e', 'e', 'k', 's')
```

Tuple with nested tuples:

```
((0, 1, 2, 3), ('python', 'geek'))
```

**Set Data Type:** - In Python, **Set** is an unordered collection of data type that is iterable, mutable and has no duplicate elements.

Initial blank Set:

```
set ( )
```

Set with the use of String:

```
{'F', 'o', 'G', 's', 'r', 'k', 'e'}
```

Set with the use of List:

```
{'Geeks', 'For'}
```

Set with the use of Mixed Values

```
{1, 2, 4, 6, 'Geeks', 'For'}
```

**Dictionary:** - Dictionary in Python is an unordered collection of data values, used to store data values like a map, which unlike other Data Types that hold only single value as an element, Dictionary holds *Key: Value* pair. Key-value is provided in the dictionary to make it more optimized.

Empty Dictionary:

```
{ }
```

Dictionary with the use of Integer Keys:

```
{1: 'Geeks', 2: 'For', 3: 'Geeks'}
```

Dictionary with the use of Mixed Keys:

```
{1: [1, 2, 3, 4], 'Name': 'Geeks'}
```

Dictionary with the use of dict():

```
{1: 'Geeks', 2: 'For', 3: 'Geeks'}
```

Dictionary with each item as a pair:

{1: 'Geeks', 2: 'For'}

#### Q4: What is an expression made up of? What do all expressions do?

**Answer: -**

An expression is a combination of operators, constants and variables. An expression may consist of one or more operands, and zero or more operators to produce a value.

- 1. Constant Expressions:** These are the expressions that have constant values only.  
**Example:**  $x = 15 + 1.3$
- 2. Arithmetic Expressions:** An arithmetic expression is a combination of numeric values, operators, and sometimes parenthesis.  
**Example:**  $x + y$ ,  $x - y$ ,  $x * y$
- 3. Integral Expressions:** These are the kind of expressions that produce only integer results after all computations and type conversions.  
**Example:** -  $a = 13$   $b = 12.0$   
 $c = a + \text{int}(b)$
- 4. Relational Expressions:** In these types of expressions, arithmetic expressions are written on both sides of relational operator ( $>$ ,  $<$ ,  $>=$ ,  $<=$ ). Those arithmetic expressions are evaluated first, and then compared as per relational operator and produce a boolean output in the end. These expressions are also called Boolean expressions.  
**Example:** -  $p = (a + b) >= (c - d)$
- 5. Logical Expressions:** These are kinds of expressions that result in either *True* or *False*. It basically specifies one or more conditions

Operator	Syntax	Functioning
and	P and Q	It returns true if both P and Q are true otherwise returns false
or	P or Q	It returns true if at least one of P and Q is true
not	not P	It returns true if condition P is false

**Q5: This assignment statements, like `spam = 10`. What is the difference between an expression and a statement?**

**Answer:-**

**Expressions** only contain identifiers, literals and operators, where operators include arithmetic and Boolean operators, the function call operator `()` the subscription operator `[]` and similar, and can be reduced to some kind of "value", which can be any Python object.

Examples: `3+5`

```
map (lambda x: x*x, range (10))
```

```
[a.x for a in some_iterable]
```

**Statements** on the other hand, are everything that can make up a line (or several lines) of Python code.

**Note:** - expressions are statements as well.

Examples: `print 42`

```
if x: do_y ()
```

```
return
```

```
a = 7
```

**Q6: After running the following code, what does the variable `bacon` contain?**

```
bacon = 22
```

```
bacon + 1
```

**Answer:** 22

**Q7. What should the values of the following two terms be?**

`'spam' + 'spamspam'`

`'spam'*3`

**Answer:** - `'spam' + 'spamspam' = 'spamspamspam'`

`'spam'*3 = 'spamspamspam'`

**Q8. Why is `eggs` a valid variable name while `100` is invalid?**

**Answer:** -

`'eggs'` -> It is a variable since it is starting with a letter.

`100` -> It is invalid because it is starting with a number and we all know that Variable names cannot begin with a number.

**Q9. 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.

**Q10. Why does this expression cause an error? How can you fix it?**

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

**Answer:** -

It will throw error because string cannot be concatenate with integer data type.

To fix this error, we have to convert the statement in single datatype i.e either in string or in integer.

Here I have converted the 99 into string variable

**Input:** - 'I have eaten' + str ("99") + 'burritos.'

**Output:** - 'I have eaten99burritos.'

\*\*\*\*\*END\*\*\*\*\*

