Variables and Arithmetic Operations

One of the most important features of any programming language is its ability to manipulate variables. A variable is just a name that refers to a value; you can think of a variable as a box that stores a piece of data.

In Python, the basic data types are strings and numbers. There are two types of numbers: integers (both positive and negative) and floats (fractional numbers with a decimal point). You can assign numbers to variables very easily. Try running the following program:

```
a = 324
b = 24
c = a - b
print ('a - b is', c)
```

In the above code, a, b, and c are all integers, and 'a - b is' is a string. The result of this program is to print:

```
a - b is 300
```

You can now use all common arithmetic operations involving numbers:

- Addition: `2 + 3 == 5`
- Subtraction: `5 - 2 == 3`
- Multiplication: `3 * 4 == 12`
- Division: `15 / 3 == 5`
- Division remainder: `18 % 5 == 3`
- Exponentiation: `2 ** 3 == 8`

It is important to note that if you try to divide two integers, Python always rounds *down* the result (so `18/5 == 3`).

To obtain a precise result for this division, you need to indicate floating point division; either of the following expressions results in a "float" data type: `18.0/5 == 3.6` or `float(18)/5 == 3.6`

In Python, the single equals sign (`=`) means "assign a value to a variable". For example, `a = 3` assigns 3 to the integer a. In order to denote equality, Python uses the double equals sign (`==`).

In Python, a string is an ordered sequence of letters, numbers and other characters. You can create string variables just like you did with:

```
a = "Hello"
b = "World"
```

Notice that the string must be surrounded by " or ' (but not a mix of both). You can use quotes inside the string, as long as you use the opposite type of quotes to surround the string, e.g., `a = "Monty Python's Flying Circus"` or `b = 'Project "Rosalind"'`.

String operations differ slightly from operations on numbers:

```
a = 'Rosalind'
b = 'Franklin'
c = '!'
print (a + ' ' + b + c*3)
```

Output:

```
Rosalind Franklin!!!
```