Variables and Constants in Python

Variables – Most students are familiar with variables in the context of mathematics.

Example, x = 6 or y = mx+b.

In programming, a variable is used in very much the same way and will reserve storage in memory to save the values of data that have the potential to change every time a program is run. The values stored in a variable can potentially change within the execution of a program.

There are four standard naming conventions for variables in programming such as camelCase, snake_case, kebab-case, and PascalCase. In this course, we will be using **camelCase**.

camelCase follows the following guidelines:

- Must begin with a lower case letter of the alphabet
- Must contain only letters of the alphabet, ending with at most one digit
- The first character of internal words are capitalized (Ex. my**N**ame)
- Must be meaningful and describe what the variable is without additional explanation

Examples of variable names: payRate, cardOne, firstName, price1, num, val, unitPrice

Constants - A constant is used to reserve storage in memory and save the values of data that do not normally change every time a program is run. The values stored in a constant cannot change within the execution of a program. Constants are typically written in all capital letters. Common examples are the value of pi (3.14159265359). The value of pi never changes thus is a constant. Other examples might be Avogadro's number from Physics and Chemistry.

Rules for forming constant names.

- Must contain only upper-case letters of the alphabet
- Multiple words in the same constant name are separated by an underscore _ character.
- Must be meaningful

Examples of constant names: TAX_RATE, MIN_VALUE, DEFAULT_SCORE

Types of Variables

A variable, to include constants, can be of several different data types. Many of these types you know by other names such as decimal, whole number, etc. In programming there are standardized data types used across all languages.

- Int (short for integer)

- An int value is a whole number. It can be positive or negative but never has a decimal value. Common examples are age or counted objects.
  - Examples: 1, 7, 12, 1500, -52
- Float
  - A float value is any number with a decimal. It can be both positive and negative
  - Examples: 1.75, 3.14, -4.7645
- Bool (short for boolean)
  - A boolean is an either/or value. Such as True or False, 0 or 1. In fact, a 0 is equal to False and True is equal to 1
  - Example: 7 = 7 is True or 6 = 3 is False
- String
  - A string is a collection of letters or numbers, often called characters. They can be numbers or letters but are considered part of a sentence. Numbers in a string cannot have arithmetic applied to them.
  - example: "The boy is walking" or "What is your favorite Color?")

In Python, variables and constants do not need to be declared before they are used in your program. The type of data stored is handled dynamically by Python. The type of data can change within the execution of a program.

The final concept is mutability. Mutability is the ability to change the value of a variable. Some variables can be changes and some cannot. An integer or float variable are considered mutable. Example, if x = 7 then you can do x = x + 7. This will change X to equal 14.

Strings are immutable. This means you cannot have a string, like "The boy walking" and add the word "is" into the sentence. The string cannot be changed. In order to change the string to add the word "is", you have to rewrite the entire string as "The boy is walking" and save it in a variable.

If the idea of mutable and immutable is confusing, don't worry. Just store the fact that strings are immutable. This is important for module 7 when we begin discussing strings in greater detail.

The variable types discussed here apply to all computer languages. However, other languages may include some other data types like char and double.