CoderZ Summer Assignment #2

Shaurya Singh

Contents

Hello everyone! This is the second assignment you will receive this summer. By this point, all of you should have received and submitted Assignment #1, which had you set up VSCode for Python programming. If you still need help, feel free to email me at shaunsingh0207@gmail.com.

In Assignment #2, you will learn the basics of Python 3 and how to use variables

1 Basics

Python is an interpreted programming language. You write `.py` files, in a text editor (such as VSCode), which are interpreted by the python interpreter, and executed. As with every language, Python has its own syntax.

Firstly, before you get started with this guide, create a `robotics1.py` file, this is where you will be writing the examples below, and executing them

1.1 Indentation

Indentation refers to the spaces at the beginning of a code line. In other languages, you usually have brackets or parenthesis telling you where a block of code starts and ends. For example, in java we have the following

```
public class Main {
  public static void main(String[] args) {
    System.out.println("Hello World");
  }
}
```

Notice the use of {} to tell you where each part of the code starts and ends. Instead of brackets, in python we rely on just indentation. For example, if we want to do an if statement, we would write something like

```
if 5 > 2:
  print("Five is greater than two!")
```

Notice the spaces in front of print. If we remove those spaces, and write

```
if 5 > 2:
print("Five is greater than two!")
```

It will result in an error. Remember the number of spaces can be any number (I choose 2), but it must remain the same throughout the python file.

1.2 Comments

As with any language, you can (and should) be commenting your code. Comments in python start with a # character, which will mark the rest of the line as a comment

```
# Compare 5 to 2
if 5 > 2:
  print("Five is greater than two!")
```

Comments can also be added to the ends of lines.

```
if 5 > 2:
  print("Five is greater than two!") # Compare 5 to 2
```

Lastly, you can have multiline comments if needed

```
"""
Line #1
Line #2
"""
if 5 > 2:
  print("Five is greater than two!") # Compare 5 to 2
```

2 Variables

A variable is a term that represents an value. You can use variables for storing information for later usage. In Python, you assign a variable a value using the = sign. You can also assign multiple variables at once

```
x = 5
y = "John"
x, y, z = "Orange", "Banana", "Grape"
```

There are also different data types that we need to be aware of. If you want to specific the data type of a variable, you can cast it.

```
x = str(3) #x is a string, and will be '3'
y = int(3) #y is a integer, and will be 3
z = float(3) #z is a float, and will be 3.0
```

You can also get the data type of a variable, using `type()`.

```
x = str(3) #x is a string, and will be '3'
print(type(x))
```

Lastly, you can define values in a list and unpack them to variables later

```
fruits = ["apple", "banana", "Grape"]
x, y, z = fruits
```

Remember that variables are case sensitive, so `John` and `john` are two different variables. Also keep in mind you can use single quotes or double quotes. In robotics, we prefer to write variables in the camelCase format, where each word except the first starts with a capital letter, like `myVariableName`.

2.1 Global Variables

Variables created outside of a function are called global variables. Global variables can be used both inside of functions and outside

```
x = "global"
def myfunc():
    print("this variable is " + x)
myfunc()
```

In this case, the variable x is a global variable

```
def myfunc():
  x = "global"
  print("this variable is " + x)

myfunc()
print("this variable is " + x)
```

In this case it isn't, and so the last print statement won't function

3 Assignment

1. Create a file named `assignment2.py`
2. Write some python code that defines a global variable, defines a function that prints `Hello World` + the variable, and call that function