ROBOLINK BASECAMP

1.5: Variables Part 1: Data Types

You might know what variables are in math or science, but what do they do in programming? This lesson will answer that question and show you how to use them.

This lesson works with:

CoDrone EDU

CoDrone EDU (JROTC ed.)

Python

beginner | CDE | CoDrone EDU | fundamentals | programming | Python | variables

Grade level:
6 - 12+

Approx. time required:
30 - 45 mins

Step 1

Goals and Steps to Success

By the end of this lesson, you should be able to:

- Define what a variable is in programming
- Explain how variables can be used in a program
- Describe the benefits of using variables in your programs
- List the different types of data in python programs
- Explain why we have different data types and provide examples of when to use the different data types

Steps to Success

1. Why variables?
2. What's in a name? Variable Naming Best Practices
3. Types of variables
4. Practice Exercises with Variables
5. Sine Waves
6. Challenge: The Drone Slide

Step 2

Types of variables

So far, we learned to store whole number values like 3 and 2, but these are completely different types of information that we might want to store. For example, how would we store the text of a social media post or keep track of its number of likes? We can use variables! Python supports a number of variable types, but we will focus on four that we will encounter most frequently in our programs.

- **Int:** positive and negative whole numbers like 10,-3, or 142
- **Float:** positive and negative decimal values like 3.14, -1.2, or 54.214
- **String:** a group of letters and/or numbers such as "Hello World"
- **Boolean**: Can be set to True or False values

There are a few important rules that you should know regarding data types of variables. First is that strings must be enclosed in single or double quotes '' or "". For these lessons we will use primarily double quotes.

Boolean values use what are called **reserved words**. These are words in a programming language that serve a specific purpose and therefore cannot be used as the names for variables and functions. The two reserved words that Boolean data types use are **True** and **False**. They must have the first letter capitalized and the rest lowercase as shown in order to be considered a Boolean value. This also means that you cannot name a variable or procedure True or False:

```
True = 2 # will result in an error
```

Going back to our social media post, we might choose to save the number of

likes as an integer data type variable while storing the text of the post as a string. We could even create a variable called hasImage which could store the Boolean data type telling us whether or not the post has an image attached. What other information might we store and as what data types?

Now that we know a little about data types, let's try creating some different types of variables. The function type() will return the data type of a variable.

```
w = 4
x = 3.2
phrase = "Hello"
is_flying = True

print(type(w))
print(type(x))
print(type(phrase))
print(type(is_flying))
```

Run this code and check the Python console. You should see each of the data types printed for each of the variables.

Step 3

Practice Exercises with Variables

Now let's try changing some of the variable values and see what happens.

```
w = 4.0
x = 3
phrase = ""
is_flying = 2>1

print(type(w))
print(type(x))
print(type(phrase))
print(type(is_flying))
```

Notice how adding or removing decimal points determines whether a variable is a **float** or an **int.**

You also see that a phrase can be empty and still be a **string**. In Python, a string is any variable in "quotation. marks".

Finally, **booleans** can also be logical statements. The statement 2>1 is a boolean because it can either be True or False. You will learn more about logical operators later. Try some of these challenges:

1. Change the value of is_flying to False
2. Create 4 variables, each representing one of the four data types we discussed
3. Try to create a variable with a phone number as its value.
    1. What happens?
    2. How would you fix this issue?
4. Try to create a variable with an email as its value.
    1. What happens?
    2. How would you fix this issue?
5. What data type do emails and phone numbers have to be stored as?
6. What data type should you store a date as?
7. Create two variables a and b. Enter the command a=b
    1. What happens?
    2. What happens when you change the value of a? b?

Step 4

Sine Waves!

Now that we have a better understanding of the purpose of variables in general, let's look at an example where their use makes a program that we have written simpler. To get set up, complete the following steps:

1. Create a new directory called 3_*variables*
2. Create a new file called 3_*1_sine*
3. Copy the code from where you wrote the the sine wave challenge

Here is an example of what the code could look like:

```
from codrone_edu.drone import *
drone = Drone()
drone.pair()
```

```
drone.takeoff()

drone.set_pitch(40)

drone.set_throttle(40)

drone.move(1)

drone.set_throttle(-40)

drone.move(1)

drone.set_throttle(40)

drone.move(1)

drone.set_throttle(-40)

drone.move(1)

drone.land()
```

Now, we will add two variables called duration and power:

```
duration = 1 # integer value for how long to fly in a direction
power = 40 # integer value for the power setting of the drone
```

We will now substitute the variables for the parameters in our flight movement commands as shown below:

```
from codrone_edu.drone import *
drone = Drone()
drone.pair()

speed = 40
duration = 1

drone.takeoff()

drone.set_pitch(speed)

drone.set_throttle(speed)
```

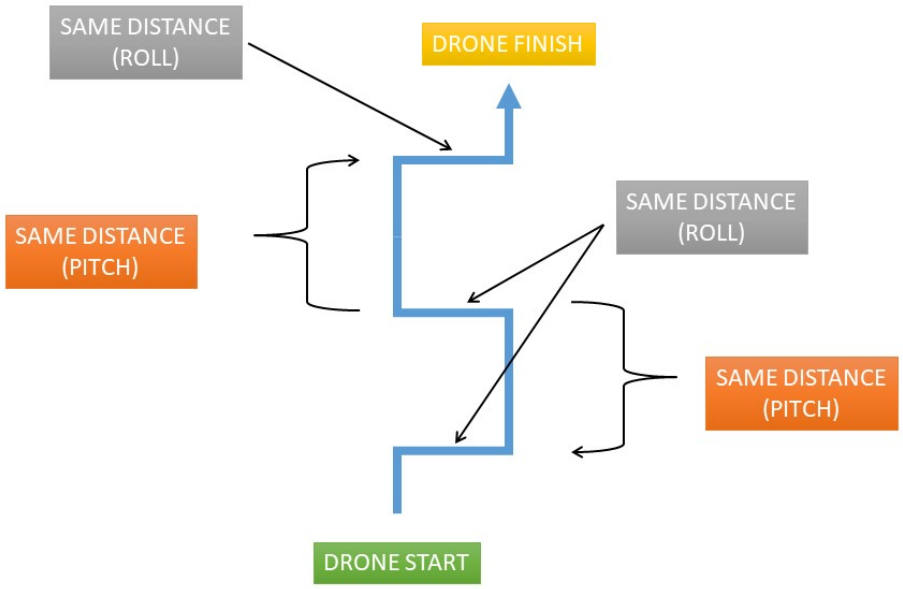
```
drone.move(duration)

drone.set_throttle(-speed)

drone.move(duration)

drone.set_throttle(speed)

drone.move(duration)

drone.set_throttle(-speed)

drone.move(duration)

drone.land()
```

You may be thinking, “That didn’t really save me any time and my code is actually a little bit longer. So why would I use variables here?” The answer is that since we are flying in a sine wave, all we have to do is change the values of the speed variable to change our sine wave’s amplitude. We can also change our sine waves period by changing the duration variable. This code is now easier to read and understand since we used names that describe what the variables are and the code is much more useful as it can create a wide variety of squares with minimal change.

Step 5

Challenge: The Drone Slide

Now that you know how to create variables, let’s use them in a program with CoDrone EDU! For this challenge, you need to have your drone fly a zig-zag pattern like the one shown below:



Rules:

1. Name the file 3_1_challenge
2. You must use at least 2 integer or float variables
3. The forward motions in the zig-zag must be the same length
4. The side to side motions in the zig-zag must be the same length

Step 6

Lesson Complete

In this lesson, you learned about variables as well as data types! Variables can come in many different forms, so make sure that you choose the right one for the job when you write your code.


Lesson Complete!