

# Introduction to Python basics: Variables and Data Types

Written by: Gisselle Ibarra



Python is one of the most user friendly coding languages to learn as a beginner. Not only is it used in industry but also fun projects such as creating video games or even your own website. To create something more complicated like mentioned above we need to first learn the basics. That is why today we will be going over two basic Python concepts: Variables and Data Types. This blog will give basic understanding of these concepts. Throughout your reading I encourage you to try out all of the examples yourself and engage with the

material if possible! In order to engage with the material we will need to use Python. Below is a link where you can download Python. If you are not able to do so it is okay! I will provide images and graphics.

Python download: <https://www.python.org/downloads/>

The first topic that we will be going over are Data Types.

## **I. Data Types**

Data Types are described as specific classifications or categories for data items. In Python, the most common 3 Data Types are: Integers, Floating Points and Strings. These are the 3 that we will be going over today.

### **a. Integers:**

Integers in Python are characterized as positive or negative whole numbers like -1,0,1,2,3. Think of integers as real numbers, no decimal points just whole numbers. Integers can not be words. Although the word 'seven' represents the

actual number 7 this is not considered an Integer.  
**Remember integers are only whole numbers.**

Ex 1.1:

```
integer = -1  
integer = 1  
integer = 0|
```

This example demonstrates integer values.

Ex 1.2:

```
integer = seven |
```

This example represents what I previously mentioned. Although the word seven represents the number 7 Python sees it as a word not an actual value.

## **b. Floating points:**

Floating points are similar to integers except these numbers contain decimal points. For example: 42.0, 3.1415, and -89.12 are all examples of Floating points.

Ex 1.3:

```
floating = 42.0  
floating = 3.1415  
floating = -89.12
```

Here we can see examples of floating point values and how they all contain decimal points. Also notice that floating points can be negative as well.

## **c. Strings:**

The final Data Types that we will be talking about are strings. Strings are described as a sequence of characters enclosed by quotation marks. These characters can be word or even numbers. The

quotation marks can either be single quotes ( `'` ) or double quotes ( `"` ) it doesn't matter which you decide to choose however you must be uniform with whichever you choose.

Ex 1.4:

```
string = 'Welcome to this blog!'
```

This is the correct way to use single quotes. You can notice that the string starts with a single quote and ends with a single quote which is what we want to do.

Ex 1.5:

```
string = 'Welcome to this blog!'
```

Python will not like this string because we do not end with the same type of quote that we started with, we aren't being uniform. The string begins with a single quote and ends with a double quote. If we start with a single quote we must end with a single quote.

Ex 1.6:

```
string = '42'
```

This is another example of a string. Although we would think that '42' is an integer that is not the case for this particular example. Since the number 42 is in quotation marks it is actually considered a string. Any numbers that are enclosed in quotation marks are considered string.

Ex 1.7:

```
string = '42.0'
```

This example is similar to the example above except the value 42.0 is considered a floating number because of the decimal. However, the same logic is applied to this example. Since the value is enclosed in single quotation marks the floating number 42.0 is actually the string '42.0'.

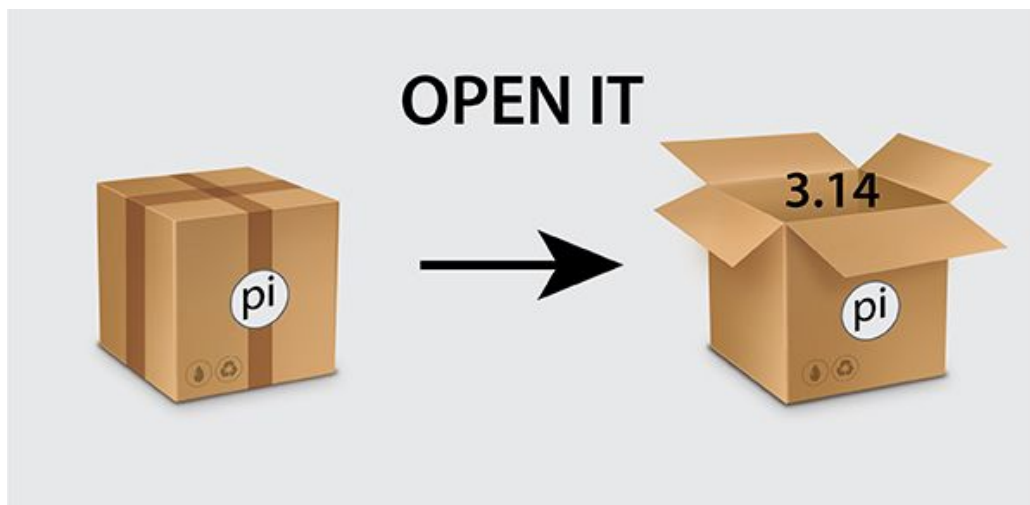
Test your knowledge:

1. 32.189 is considered what kind of Data Type?
2. '90' is a number. True or False?

3. "Hello, what is your name?" is an acceptable string. True or False?
4. -78 is what kind of Data Type?

## **II. Variables**

Now that we have learned about Data Types we will dive into variables. I want you to think of Python variables like a box. The box itself is the variable and you can place a single value into the box. The values stored inside can be any of the Data Types that we previously talked about (Integers, Floating numbers or Strings)



*(In this picture, the box represents the variable named pi. Inside the variable pi we have stored the value 3.14.)*

#### **a. Naming a variable:**

Naming a variable is important because it gives context to the store valuable that it represents. A variable can contain special characters, numbers and letters. The only thing to remember when naming a variable is that it cannot begin with a number or contain python keywords(refer to chart below). Once



you name a variable you assign it to a single value using an equal sign. To the left of the equal sign is the variable name and to the right is the value stored in that variable.

### Python keywords:

and	as
assert	async
await	break
class	continue
def	del
elif	else
except	false
finally	for
from	global
if	import
in	is
lambda	none
nonlocal	not
or	pass
raise	return
true	try
while	with
yield	

Ex 2.1:

```
cost = 6
```

This is a correct way of naming a variable. Here we have assigned the variable name `cost` to the number 6 representing the cost of an item. The variable

name cost is to the left of the equal sign and the value itself is to the right of the equal sign.

Ex 2.2:

```
6_price = 32
```

Python would not accept this variable name because it doesn't follow the naming rules. Remember what we had discussed earlier! When naming a variable we can't begin with a number. Instead maybe we could write out the number 6 into six which would be acceptable.

Ex 2.3:

```
min = 2|
```

This is another variable name that Python will not accept. The variable name is min which is a Python keyword. If you need to please refer back to the Python keyword chart.

## **b.more variables:**

Remember that variables can be assigned to any kind of Python Data Type (integer, floating number, string).

Ex 2.4:

```
number = 11|
```

Here we have assigned the variable numbers to the integer 11.

Ex 2.5:

```
price = 123.456
```

Here we have assigned the variable price to the value 123.456. Here we reflect the price of a specific item as 123.456.

Ex 2.6:

```
welcome = 'Welcome to this blog!'
```

This is an example of a variable being assigned to the string Welcome to this blog.

### Test your knowledge:

1. `7_teen = 17` is an acceptable variable name.  
True or False?
2. Write an acceptable variable name.
3. We can use Python keywords as variable names.  
True or False?

## III. Closing remarks

Learning Python or any other coding language may be difficult but hopefully after reading this blog you leave with an understanding of some Python basics. Although these skills will not fully equipt you to create your own video game right now I would say that you are going in the right direction. Data Types and Variables are just scratching the surface of what you can do with Python. Dig deeper into Python by learning more key basics!

Here is a recap of what you've learned today. There are 3 popular Data Types: Integers, floating points and strings.

- a. Integers: These are whole numbers that can be either positive or negative.
- b. Floating points: These are numbers with decimals that can also be positive or negative.
- c. Strings: These are a sequence of characters enclosed with single quotes or double quotes.

Variables relate to Data types. Variables are like a box. This box stores a single variable inside of it either an integer, a string or a floating point. The box itself would be the variable name and what's inside the box is a value.

I hope that after this blog you decided to dig deeper into Python or any other coding language. Good luck and happy coding!