Python Basics PT1

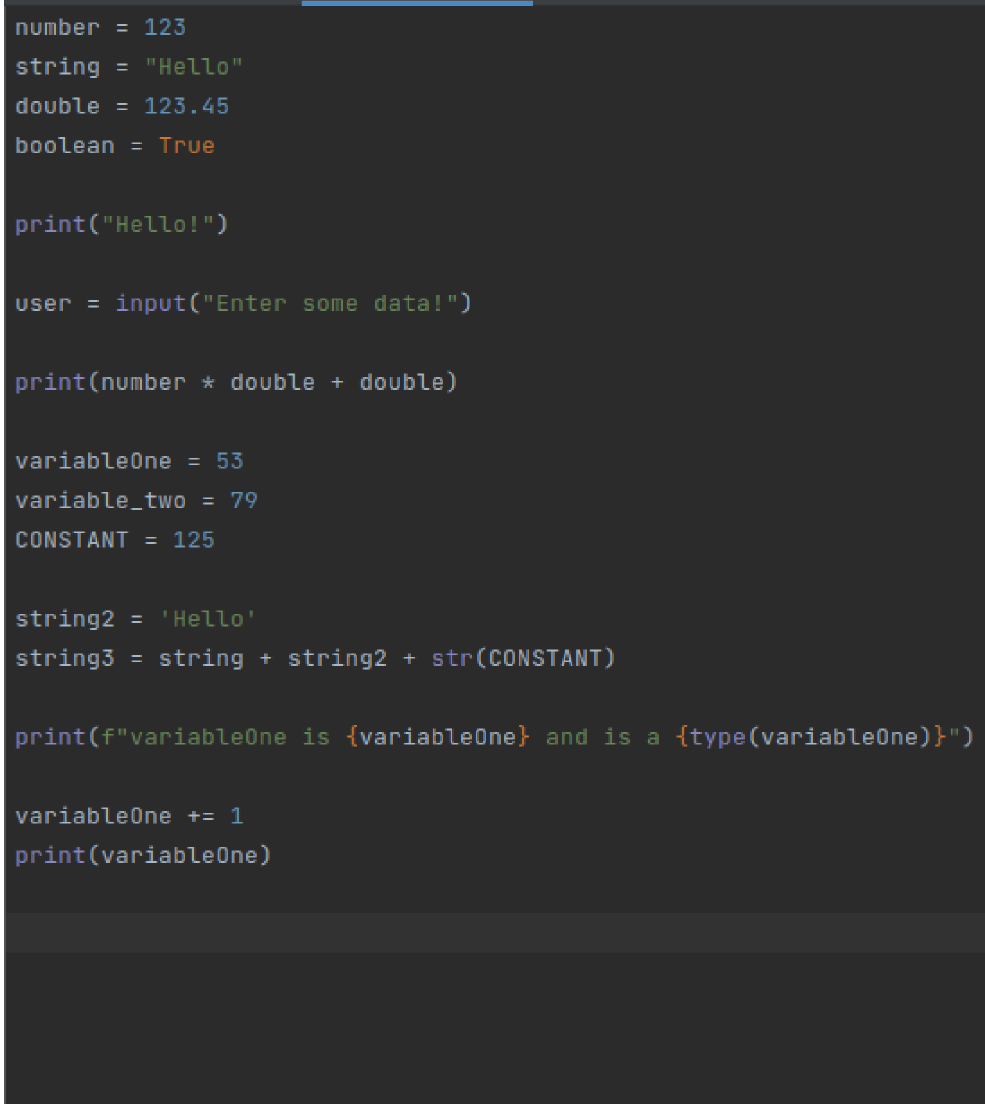
Research Data Services

By Avery Fernandez

Plans For Today

Learning the basics to python

- **Hello World** first Program
- **Variables** what they are and how they work
- **Input/Output** how to get user data and display data
- **Variable Operations** basic things you can do with variables

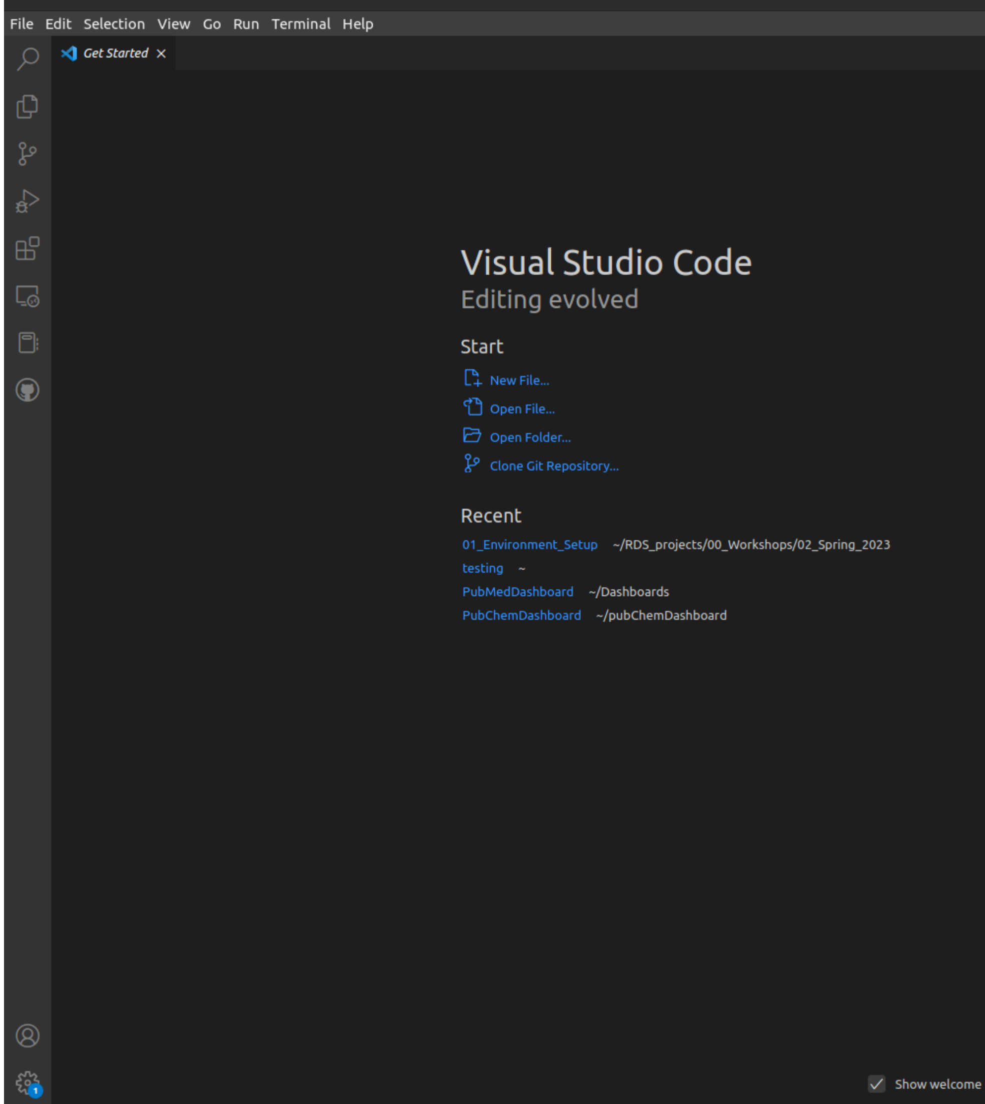


```python
number = 123
string = "Hello"
double = 123.45
boolean = True

print("Hello!")

user = input("Enter some data!")

print(number * double + double)

variableOne = 53
variable_two = 79
CONSTANT = 125

string2 = 'Hello'
string3 = string + string2 + str(CONSTANT)

print(f"variableOne is {variableOne} and is a {type(variableOne)}")

variableOne += 1
print(variableOne)
```

First Program

Steps:

1. Click **Open Folder…**
2. Create a new folder and name it
3. Click **Open**
4. Now Click **New File…**
5. Name it anything you wish, just add **.py** to the end



Hello World Program

```
print("Hello World!")
```

1. Now click the **Play** button on the top right or press Shift + Enter

Terminal Output

```
Hello World!
```

What Are Variables?

Variables are kind of like cabinets and drawers

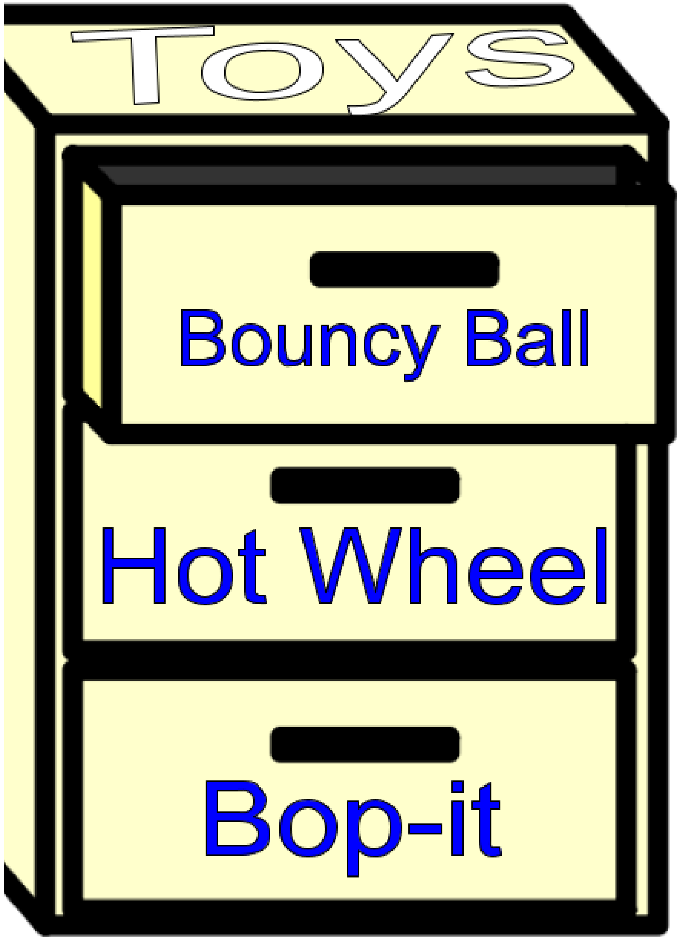
Imagine that we have 4 cabinets in front of us

Each cabinet can only have one type of item

But each drawer in the cabinet only has one item in it

If we were to label each drawer with a name, that is how variables work

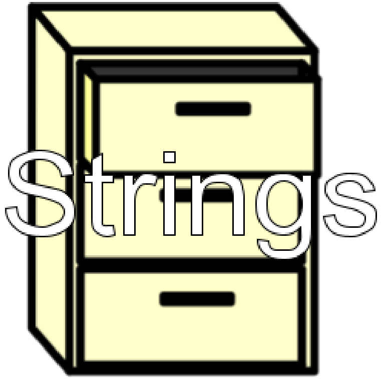
What is in the *Hot Wheel* drawer?

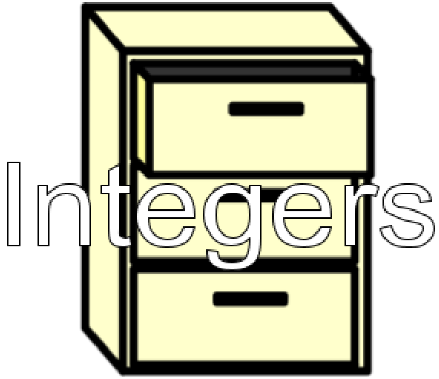


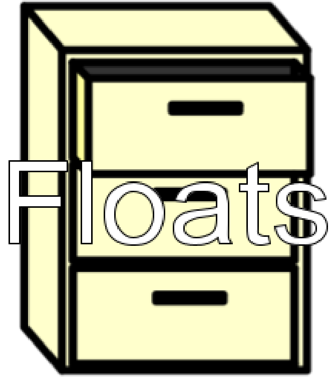
Variables Types

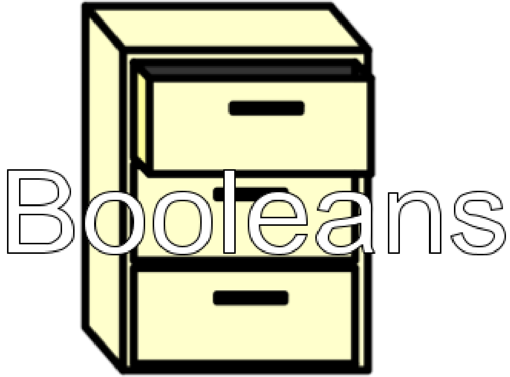
We have 4 main variable types we work with:

- Strings
- Integers
- Floats
- Booleans









Strings

Strings in python are basically words and text

Anything can be a string from a name to a word

Strings are usually denoted by being surrounded by a single ’ or double "

```
"Apollo"
```

```
'Or'
```

Integers and Floats

Integers and Floats are purely numbers

The only difference is, one is a whole number and one is a decimal number

Integers will always be just a whole number

```
10
12312412
```

While floats will always have a decimal point

```
136.93
12512562134.0
```

Booleans

They are the logic behind our madness, literally

Booleans will simply be either True or False

```
True
False
```

> ***NOTE:*** They must be capitalized

Creating a variable

Variable Name

Before we create a variable, we must give the variable a name

There are 3 ways of naming a variable

Camel Case

```
variableOne
```

Snake Case

```
variable_two
```

Constants

```
SPEED_OF_LIGHT
```

Creating a variable

Our First Variable

Now the notation for creating variables is simple

```
{variableName} = {value}
```

So let’s create some variables

```
name = "Jonathan"
age = 23
bankAccount = 147.45
currently_employed = False
```

Inputting data

Now what if we want to put in data ourselves or have others do it

We don’t want them to have to go in and edit the file

Let’s allow them to easily input data, using Python’s **input** function

Notation

```
{variable} = input({Some String})
```

Python

```python
user = input("Enter some data:")
```

Outputting data

Now that we know how to create variables and get user data

Let’s output it to the terminal

For that we use the **print** function

Notation

```
print({variable or value})
```

Remember we can either print values or variables

```
name = "Jonathan"
print("Hello")
print(name)
```

Terminal

```
Hello
Jonathan
```

Exercise One

Create 3 variables of different types

Create another variable that takes the user’s input

Then print out all 4

Exercise One Code

```
fruit = "Orange"
temperature = 85.3
canDrive = True
name = input("What is your name?")
print(fruit)
print(temperature)
print(canDrive)
print(name)
```

Terminal Output

```
What is your name?Robert
Orange
85.3
True
Robert
```

Variable Operations

So now that we have variables, what can we do with it

Checking Type

Now that we have variables, we can easily change what type they are

First let’s see how to check what type it is using the **type** function

Notation

```
print(type({variable}))
```

Python

```
number = 5
print(type(number))
```

Terminal

```
<class 'int'>
```

This means it is an Integer

Casting Variables

What if we have a number in a string, but want it as an integer

This is called casting a variable, and all you do is specify the type you want

Notation

```
str()
int()
float()
bool()
```

Example

```
number = 5
print(type(number))
numberChanged = str(number)
print(type(numberChanged))
```

Terminal

```
<class 'int'>
<class 'str'>
```

Math Operations

Programming has a lot to do with basic math

- Add +
- Subtract -
- Multiply *
- Divide /
- Integer Division //
- Remainders %
- Exponents **

Using Math Operations

```
a = 8
b = 6
print(a + b) # add
print(a - b) # subtract
print(a * b) # multiply
print(a / b) # divide
print(a // b) # floor division, rounds down
print(a % b) # modulo, returns remainder
print(a ** b) # power, exponents
```

Terminal Output

```
14
2
48
1.3333333333333333
1
2
262144
```

Exercise Two

Create a python program that takes a number from the user

Then plugs the number into the equation

$$f(x) = \frac{12x^4 - 4x^2 + 9}{x^5 - 31}$$

and outputs the results

Exercise Two Code

```python
x = int(input("Enter a number"))
upper = 12*x**4 -4*x**2 + 9
lower = x**5-31
print(upper/lower)
```

Terminal Output

```
Enter a number4
3.0382678751258814
```

Incrementation

Incrementation allows you to apply operations to a variable itself

```
a = 2
a = a + 3
print(a)
```

Terminal

```
5
```

With Incrementation

```
a = 2
a += 3
print(a)
```

Terminal

```
5
```

This can be done for any operations

String Concatenation

Combine strings together

```
greeting = "Hello"
name = "Jimmy"
sentence = greeting + ", " + name
print(sentence)
```

Terminal

```
Hello, Jimmy
```

String Formatting

Allows us to combine things in line for all variable types

```python
name = "Ezreal"
temperature = 97.4
statement = f"Hello, {name}! Have an amazing day! It is {temperature} degrees outside."
print(statement)
```

Terminal

```
Hello, Ezreal! Have an amazing day! It is 97.4 degrees outside.
```