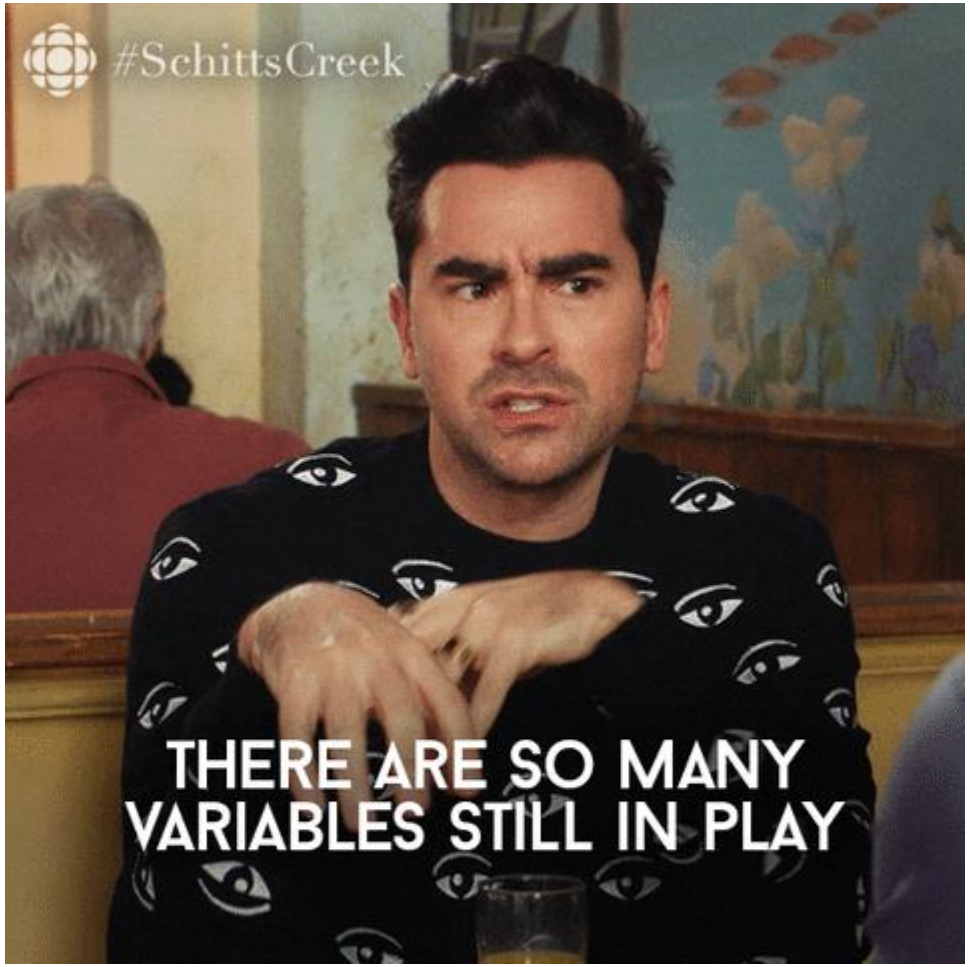
Variables

One man’s constant is another man’s variables

Alan J Perlis

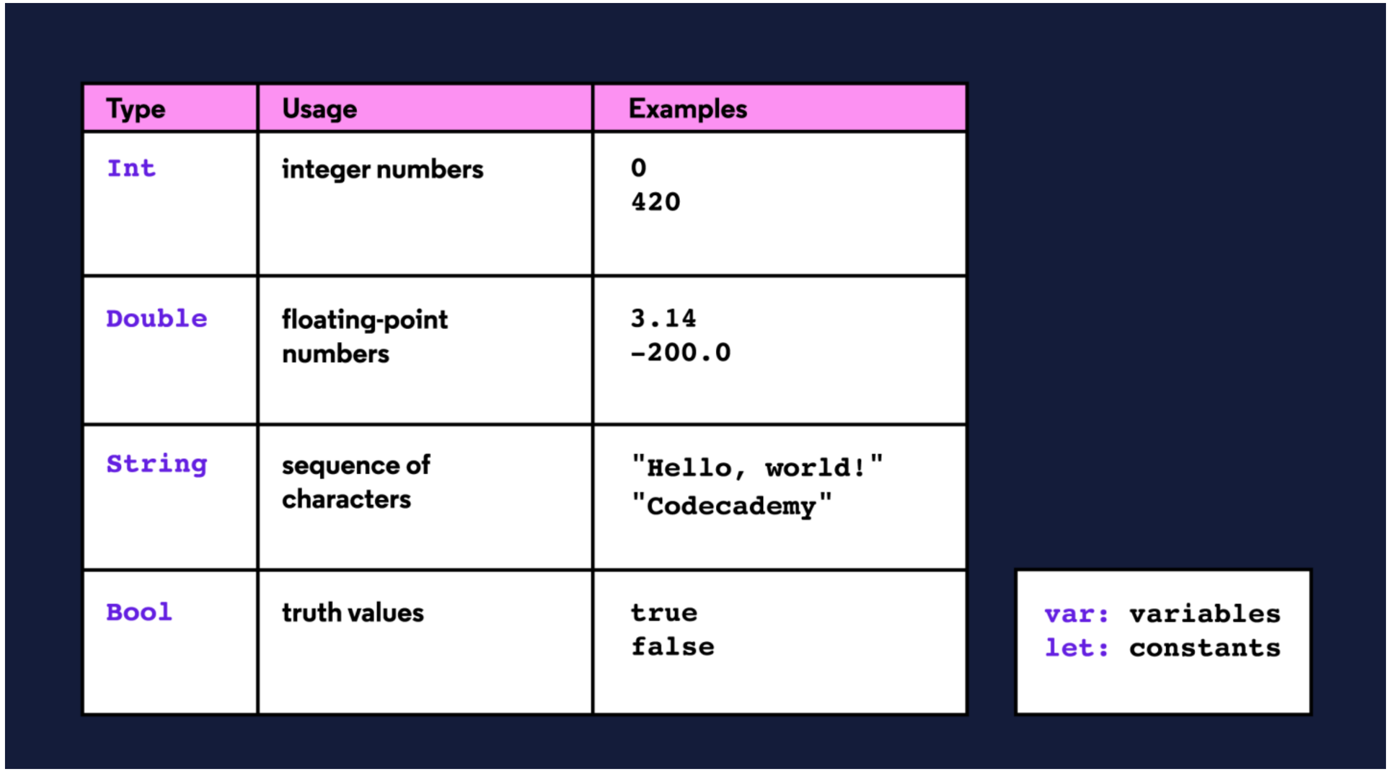
What is a variable?

- Name for a Memory location.
- Open bucket or box.
- Can change anywhere on the execution.



Types of Variables

- Integer
- Float
- String
- Bool



Type	Usage	Examples

`Int`	integer numbers	`0`
		`420`

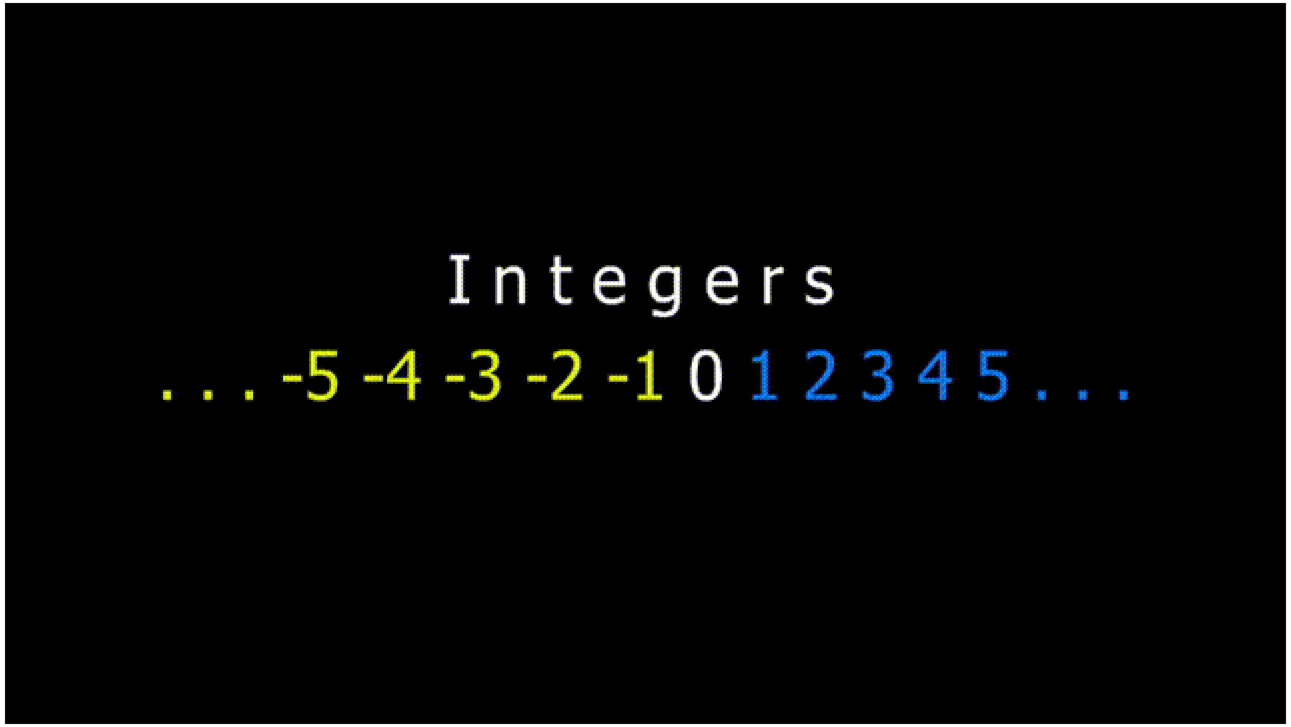
`Double`	floating-point numbers	`3.14`
		`-200.0`

`String`	sequence of characters	`"Hello, world!"`
		`"Codecademy"`

`Bool`	truth values	`true` `false`

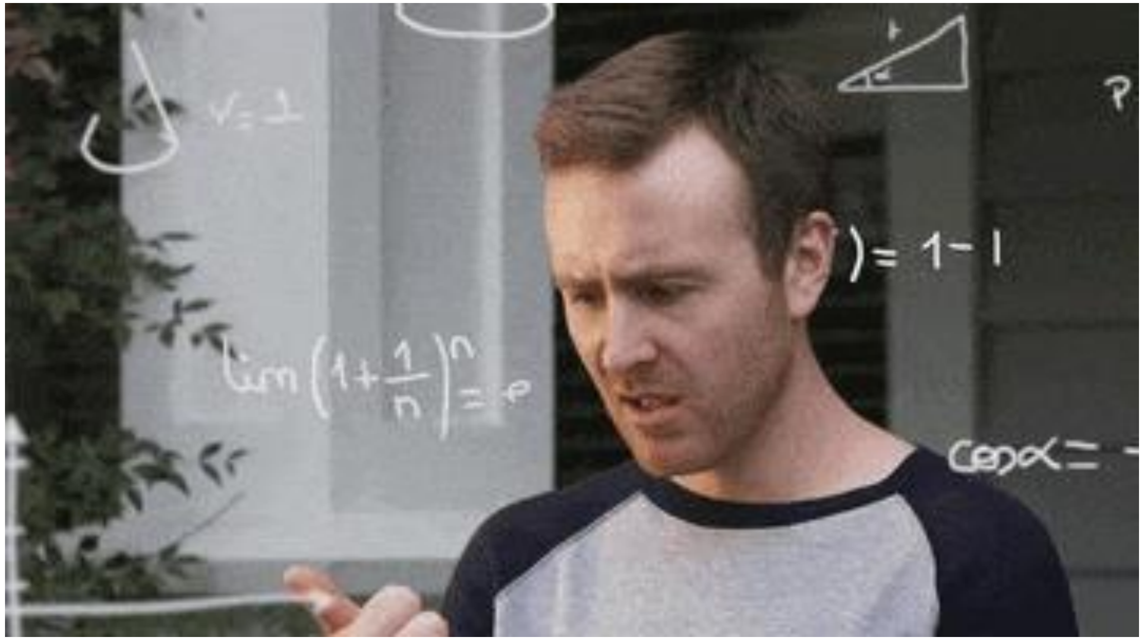
```
var: variables
let: constants
```

Integer Datatype



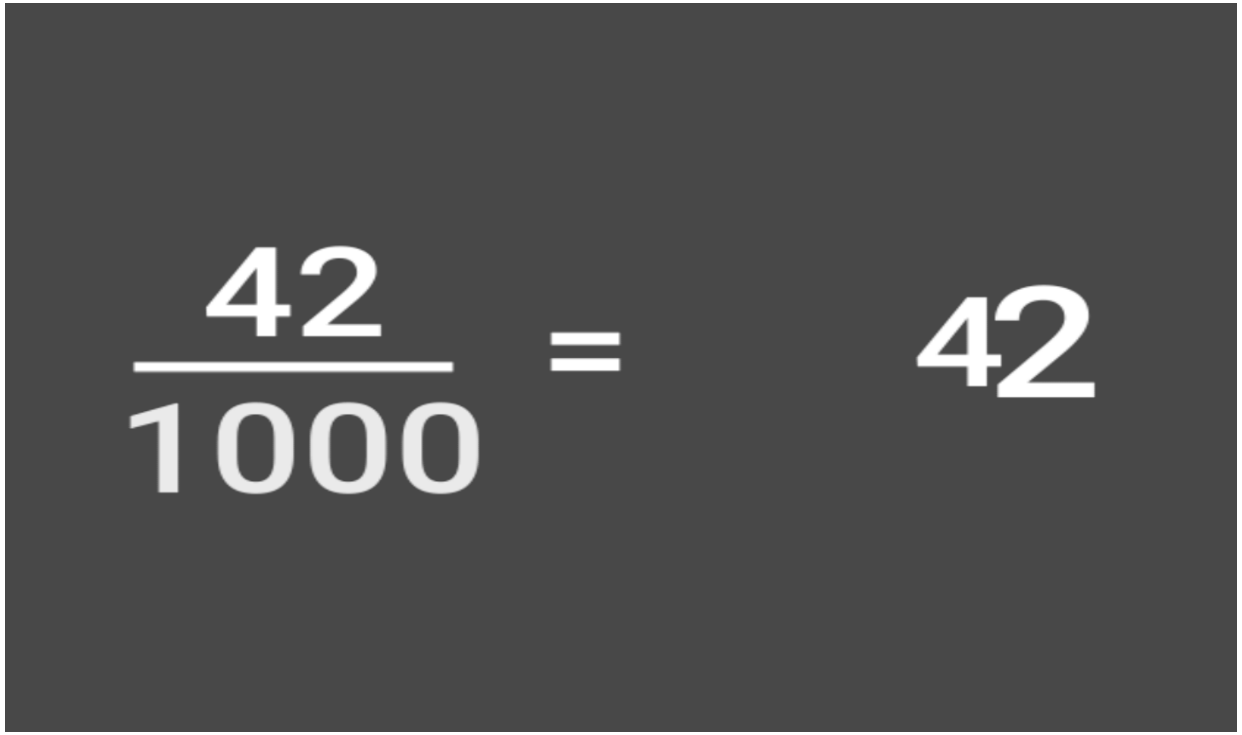
Data type	Purpose	Examples
Integer (4 bytes)	To store Integer numbers or values.	1, 10, 55, 65, 100

Long int Datatype



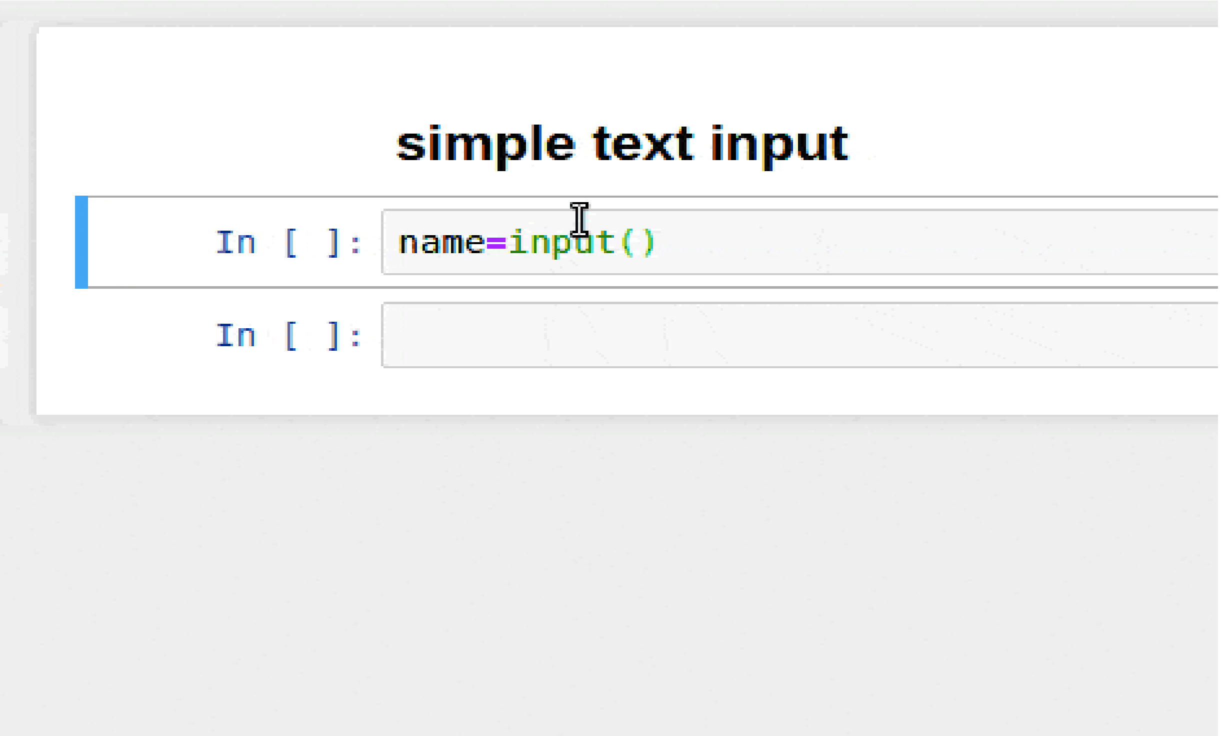
Data type	Purpose	Examples
Long Integer (8 bytes)	To store long Integer numbers or values.	1, 10, 55, 65, 100

Floating-point Datatype



Data type	Purpose	Examples
Floating point (4 bytes)	To store Floating point numbers or values.	1.2, 10.56, 55.89, 65.70, 100.59

String Datatype



simple text input

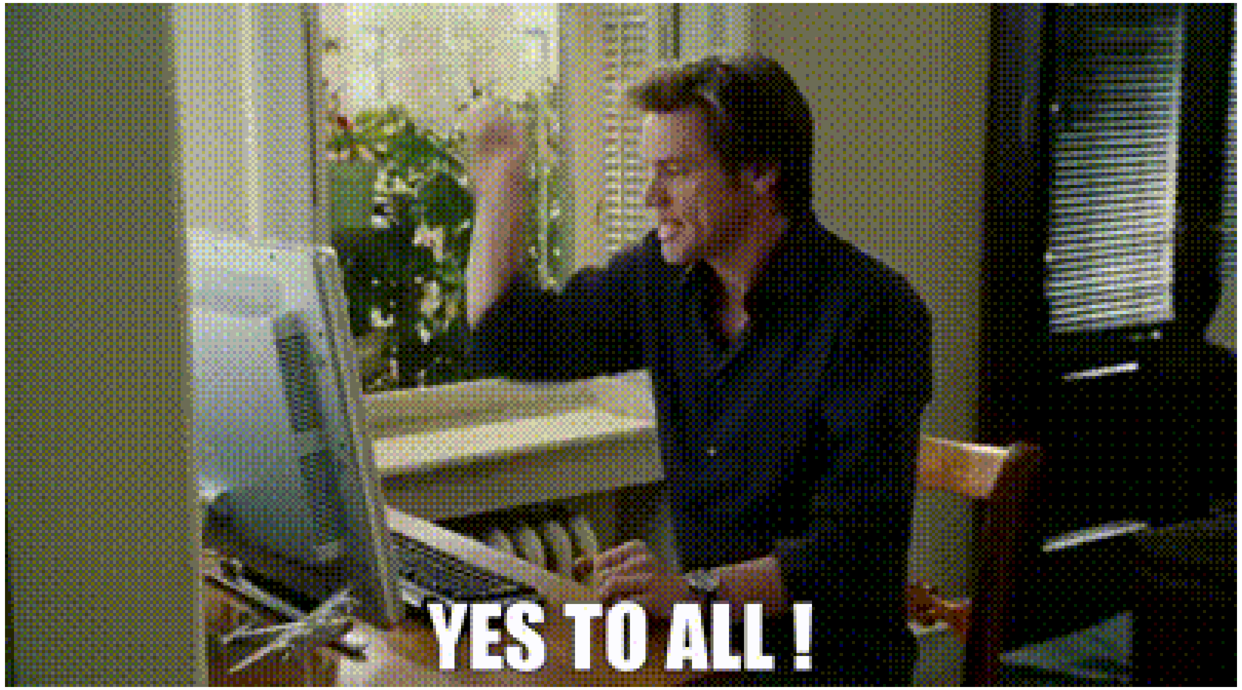
```
In [ ]: name=input()
In [ ]:
```

Data type	Purpose	Examples
String (1 char = 1 bytes)	To store string values	“hello”, “Super”, “hello12”, “35”

Boolean Datatype

Data type	Purpose	Examples
Boolean (1 bytes)	To store Boolean values	True, False

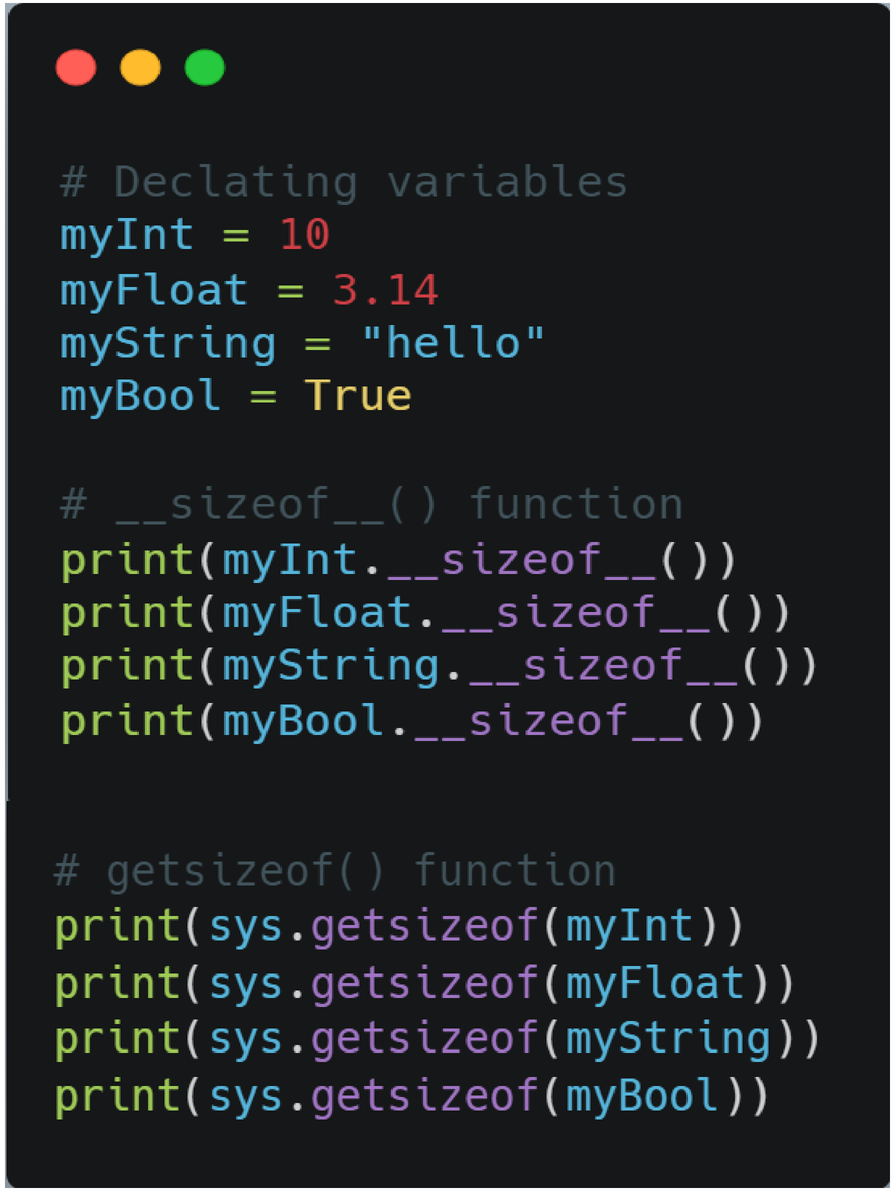
Dynamic Typing



- Does not need to explicitly declare datatype of variables.
- Simple as Yes to all and identifies the type itself.

getsizeof() vs __sizeof__()

- getsizeof() - Return the size of an object in bytes.
- __sizeof__() - Returns the internal size in bytes for the given object

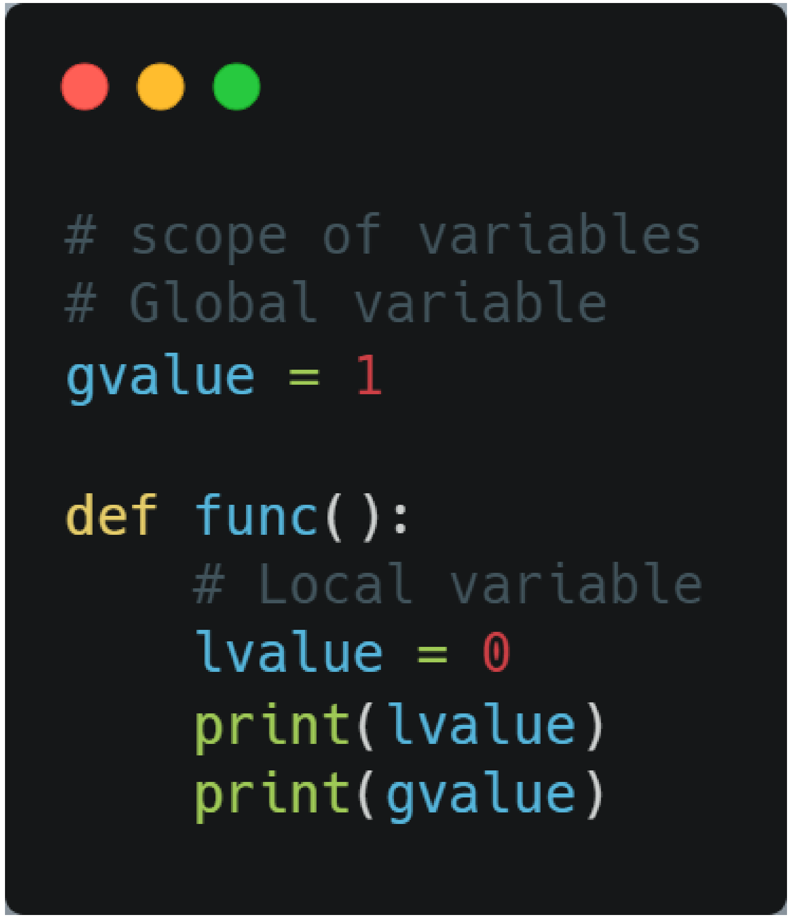


```
# Declating variables
myInt = 10
myFloat = 3.14
myString = "hello"
myBool = True

# __sizeof__() function
print(myInt.__sizeof__())
print(myFloat.__sizeof__())
print(myString.__sizeof__())
print(myBool.__sizeof__())


# getsizeof() function
print(sys.getsizeof(myInt))
print(sys.getsizeof(myFloat))
print(sys.getsizeof(myString))
print(sys.getsizeof(myBool))
```

Scope

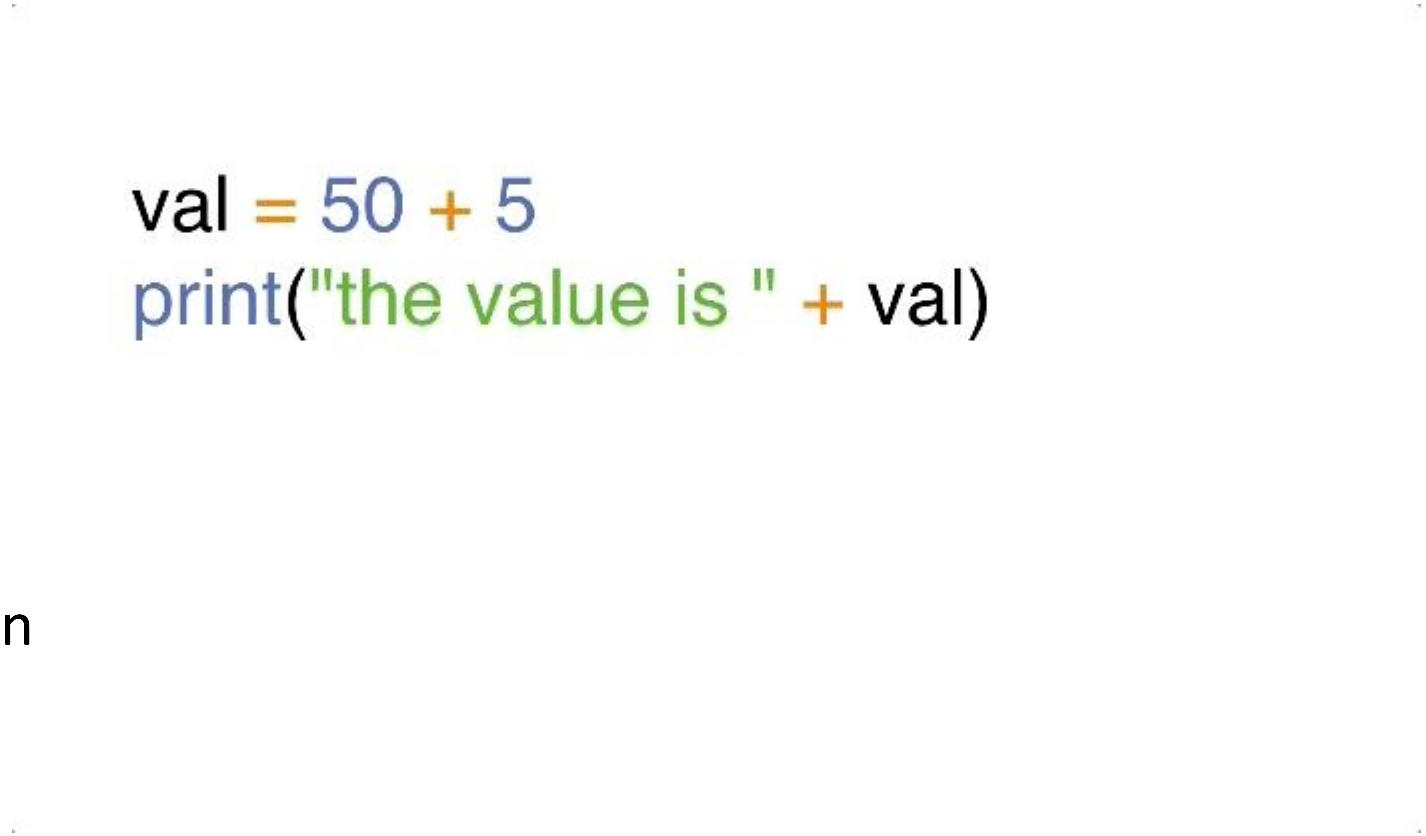


```python
# scope of variables
# Global variable
gvalue = 1

def func():
    # Local variable
    lvalue = 0
    print(lvalue)
    print(gvalue)
```

- Local variable - Can access only inside the scope in this case function.
- Global variable – Can access from anywhere of the program.

Type casting



```
val = 50 + 5
print("the value is " + val)
```

- Type Conversion can be needed in some cases.
- Some of the functions for typecasting are, str(), float(), and int().

Variables

DEMO