1

Python Quick Tips

Data Types

Future Videos

MAKING A CUSTOM NODE

MAKING A CUSTOM PACKAGE

PYTHON FUNDAMENTALS

PYTHON IN DYNAMO (SERIES)

TLDR
github

https://github.com/aussieBIMguru

Python Quick Tips #1

Data Types and variables

Data Types

In Python, Data is referred to as **objects**

Data Types

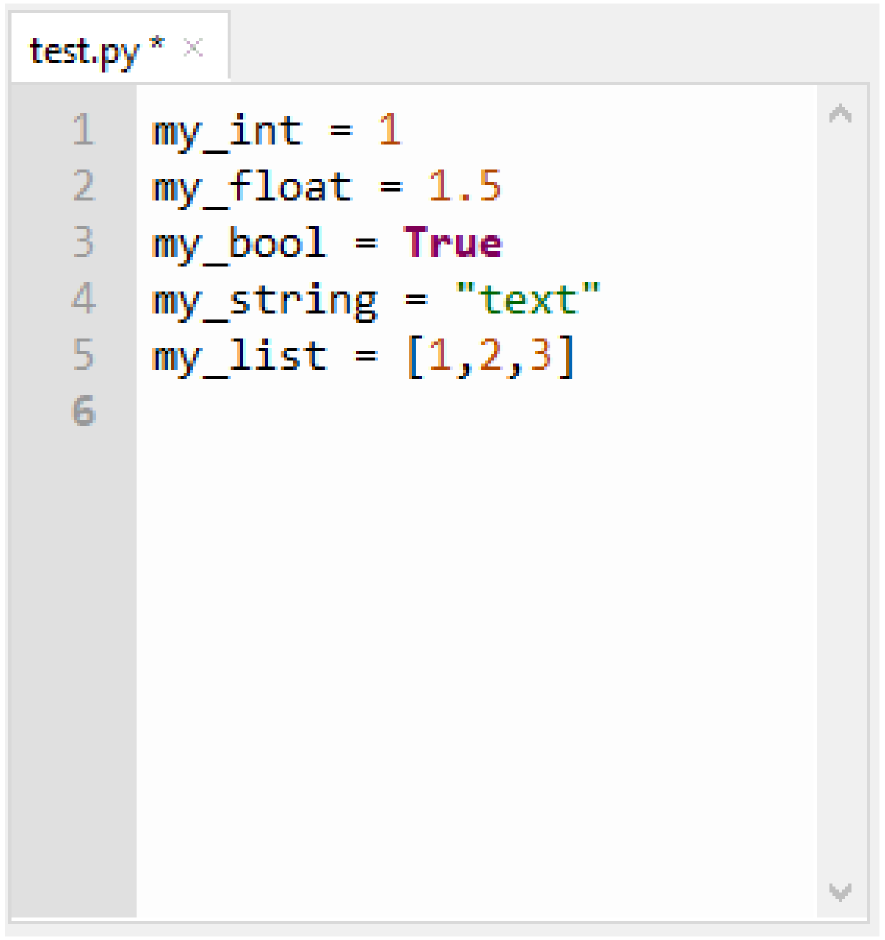
Data comes in **types**

Integers	int	1
Floats	float	1.5
Booleans	bool	True
Strings	str	“text”
Lists	list	[a,b,c]
Others	varies	varies

Declaring Variables

Syntax

Variable = value

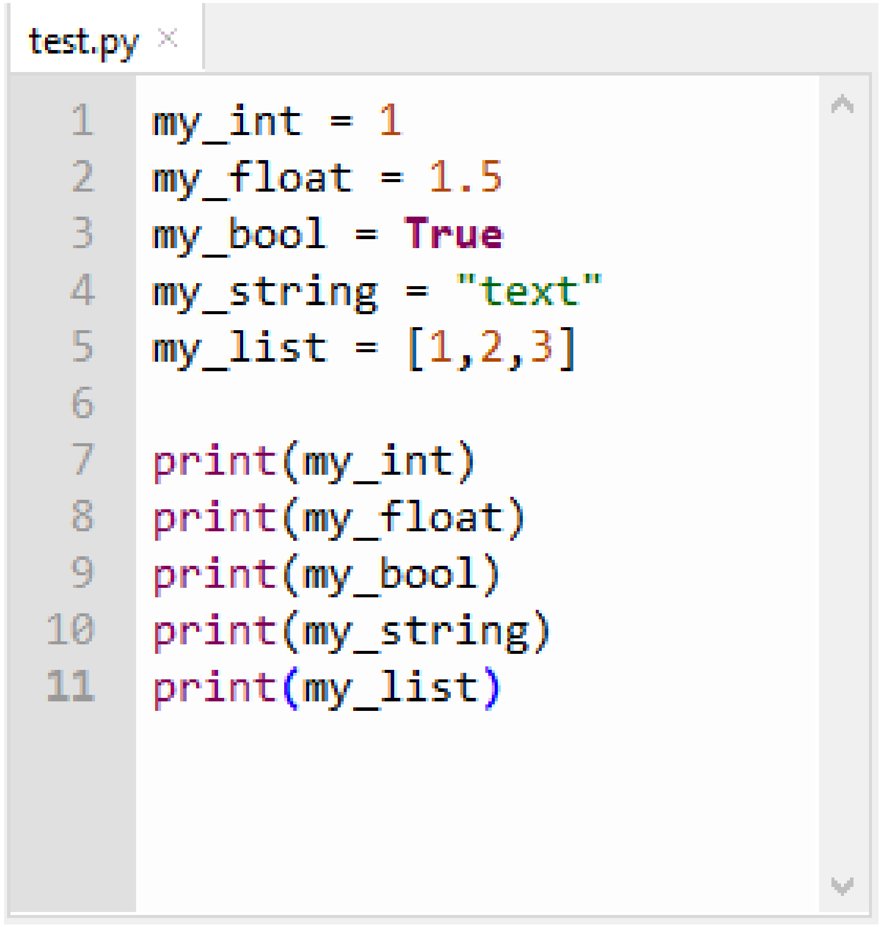


```python
my_int = 1
my_float = 1.5
my_bool = True
my_string = "text"
my_list = [1,2,3]
```

Calling/Printing Variables

Function

print(variable)

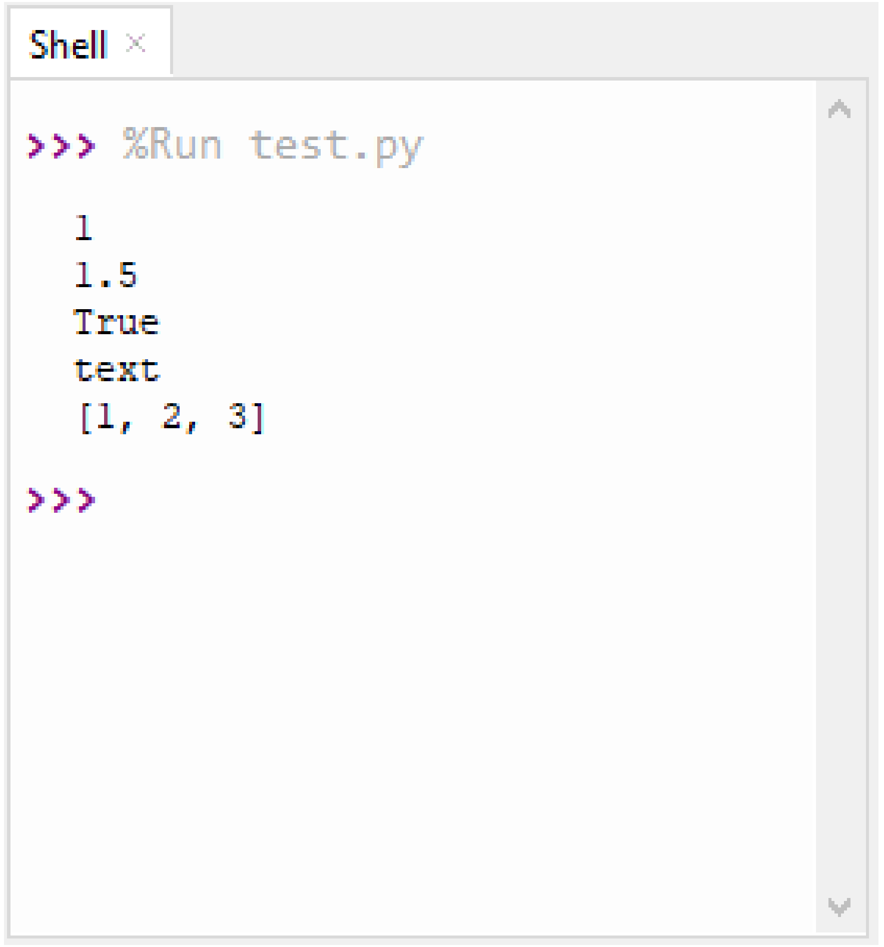


```
my_int = 1
my_float = 1.5
my_bool = True
my_string = "text"
my_list = [1,2,3]

print(my_int)
print(my_float)
print(my_bool)
print(my_string)
print(my_list)
```

Calling/Printing Variables

Result

print(variable)

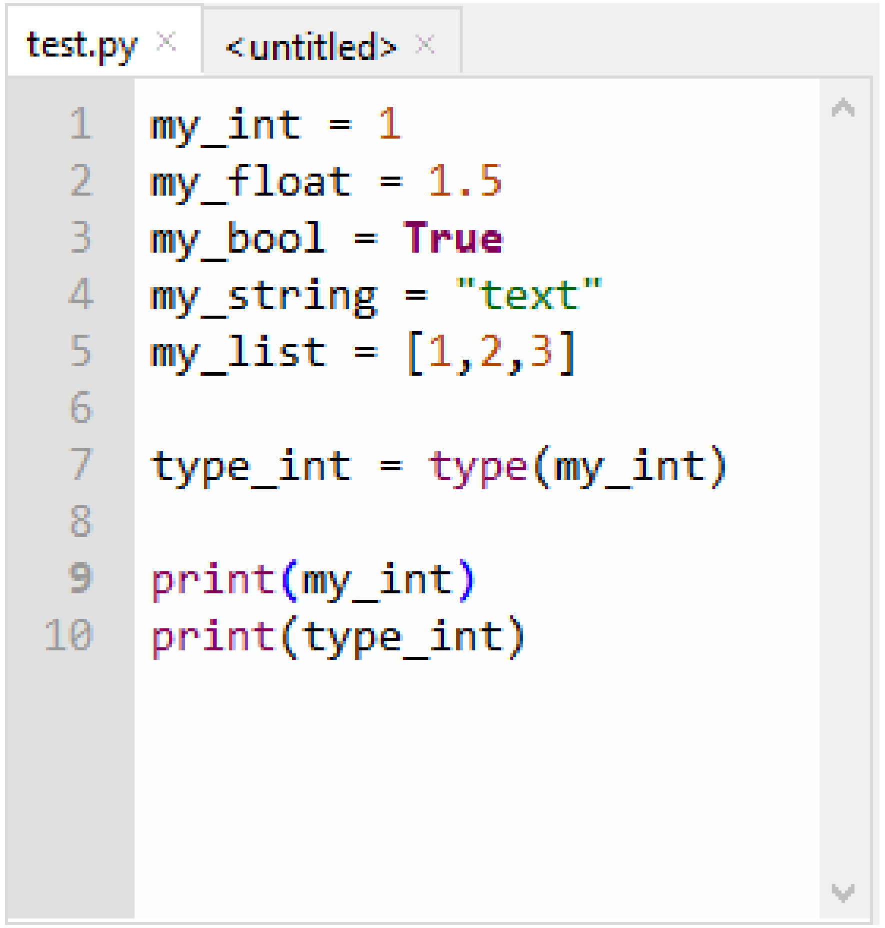


```
>>> %Run test.py

  1
  1.5
  True
  text
  [1, 2, 3]

>>>
```

Checking Data Types

Function

Type(variable)

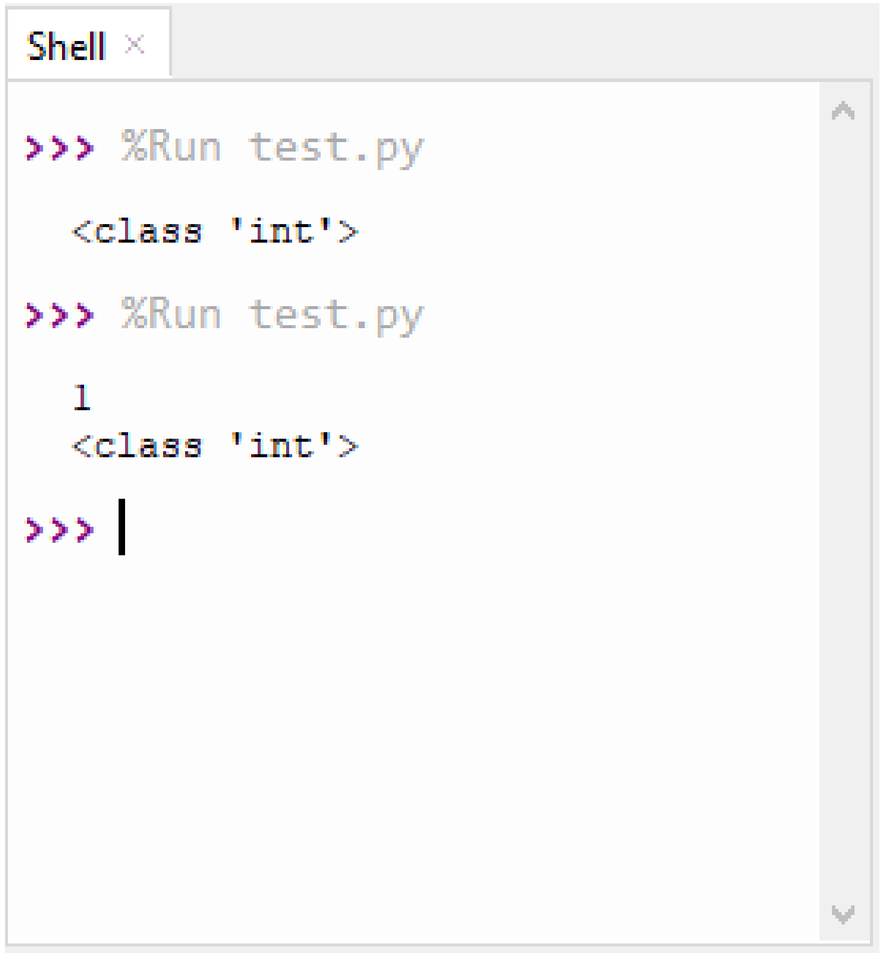


```
my_int = 1
my_float = 1.5
my_bool = True
my_string = "text"
my_list = [1,2,3]

type_int = type(my_int)

print(my_int)
print(type_int)
```

Checking Data Types

Function

Type(variable)



```
>>> %Run test.py

  <class 'int'>

>>> %Run test.py

  1
  <class 'int'>

>>> 
```

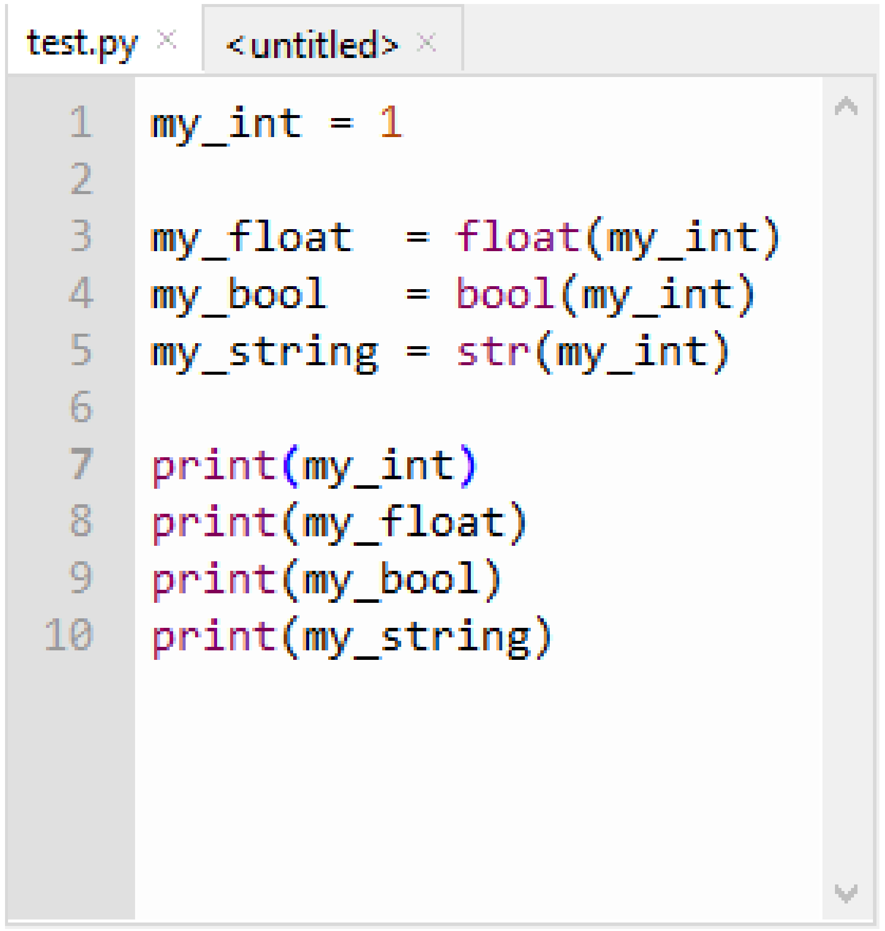
Converting Data Types

In Python, **always** deal with the right data **type**

Converting Data Types

Function(s)

class(variable)

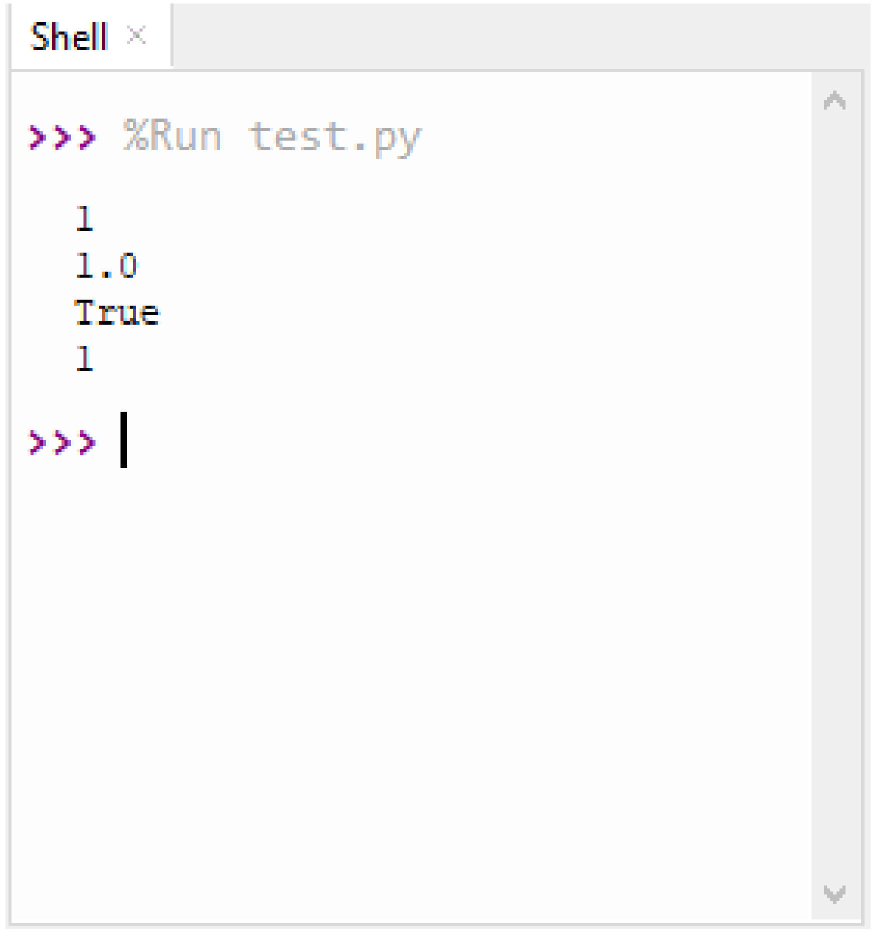


```
my_int = 1

my_float  = float(my_int)
my_bool   = bool(my_int)
my_string = str(my_int)

print(my_int)
print(my_float)
print(my_bool)
print(my_string)
```

Converting Data Types

Function(s)

class(variable)

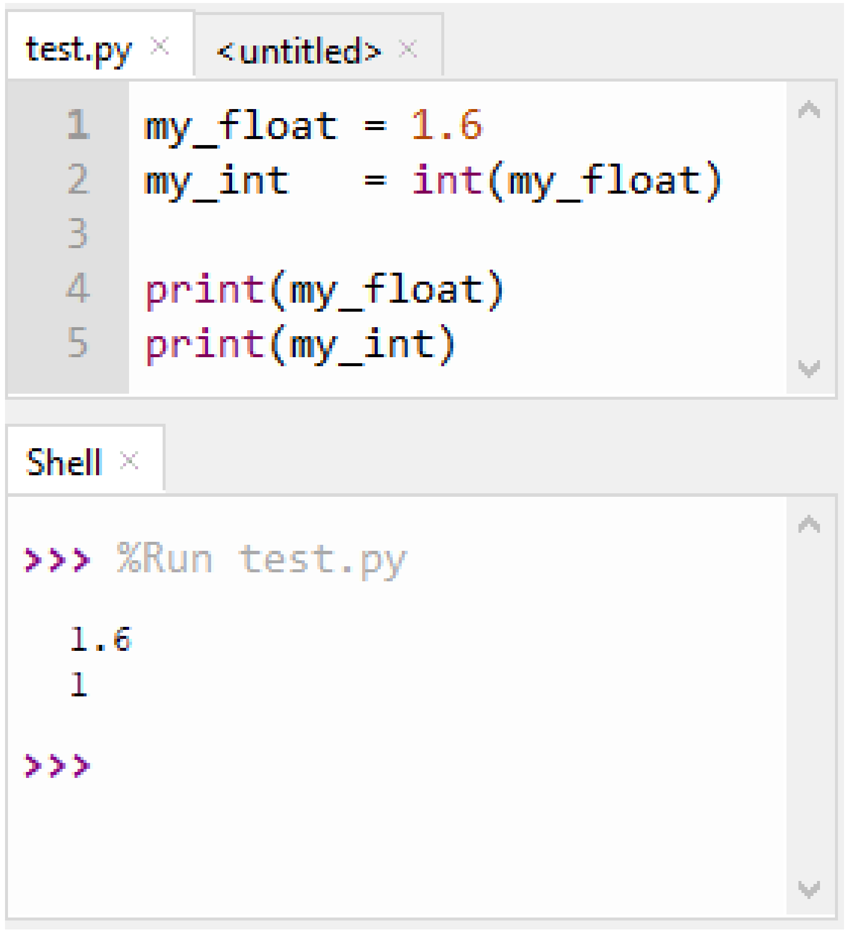


Shell ×

```
>>> %Run test.py

  1
  1.0
  True
  1

>>> 
```

Converting Data Types

Not all data types are cleanly converted...

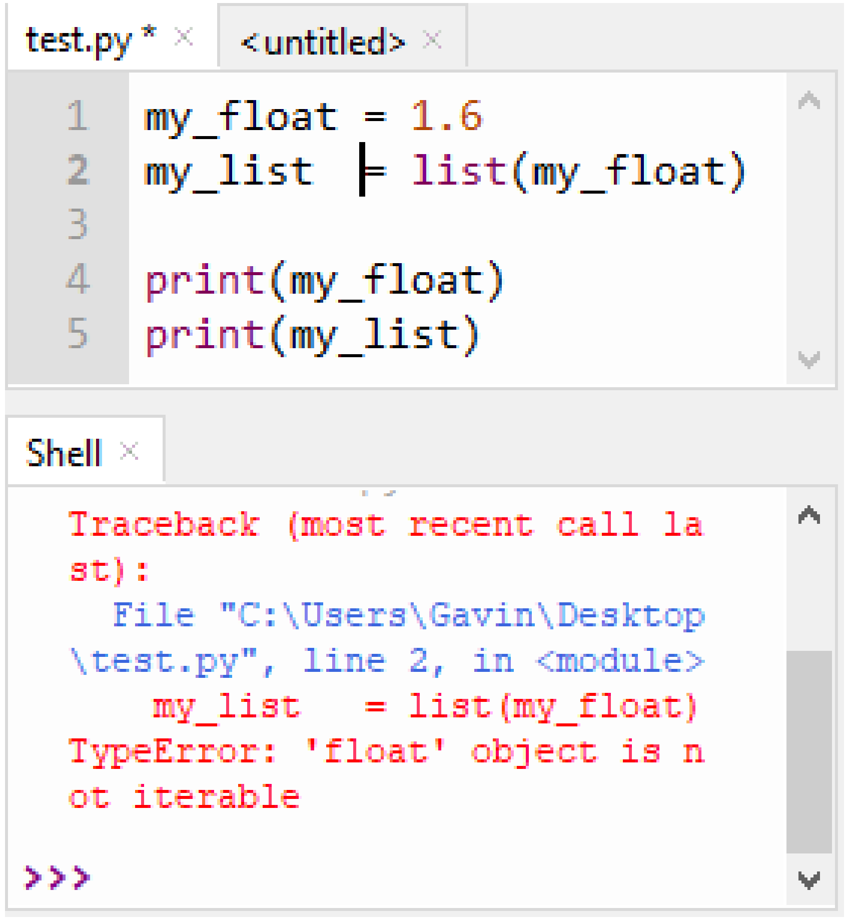


```
my_float = 1.6
my_int   = int(my_float)

print(my_float)
print(my_int)
```

```
>>> %Run test.py

  1.6
  1

>>>
```

Converting Data Types

Not all data types are able to be converted…



```
my_float = 1.6
my_list  = list(my_float)

print(my_float)
print(my_list)
```

```
Traceback (most recent call la
st):
  File "C:\Users\Gavin\Desktop
\test.py", line 2, in <module>
    my_list   = list(my_float)
TypeError: 'float' object is n
ot iterable

>>>
```

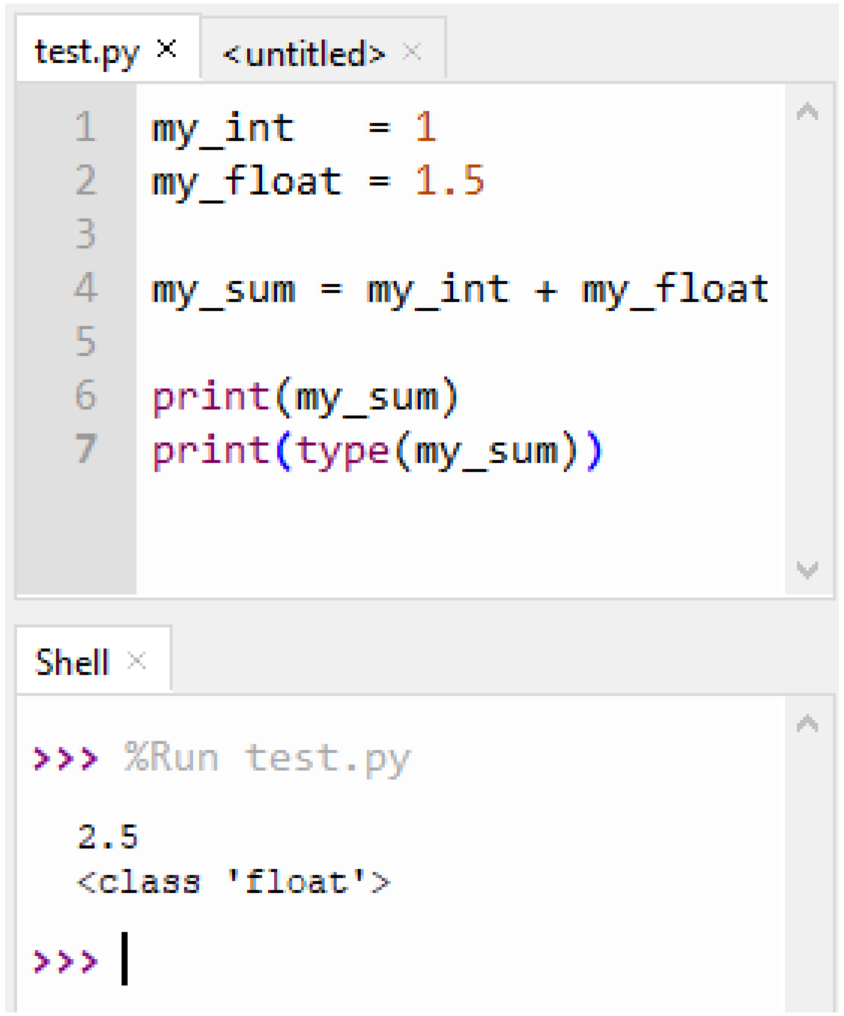
Arithmetic

Operators

Add	+
Subtract	-
Multiply	*
Divide	/
Exponential	**
Remainder	&

Arithmetic

Combining variables

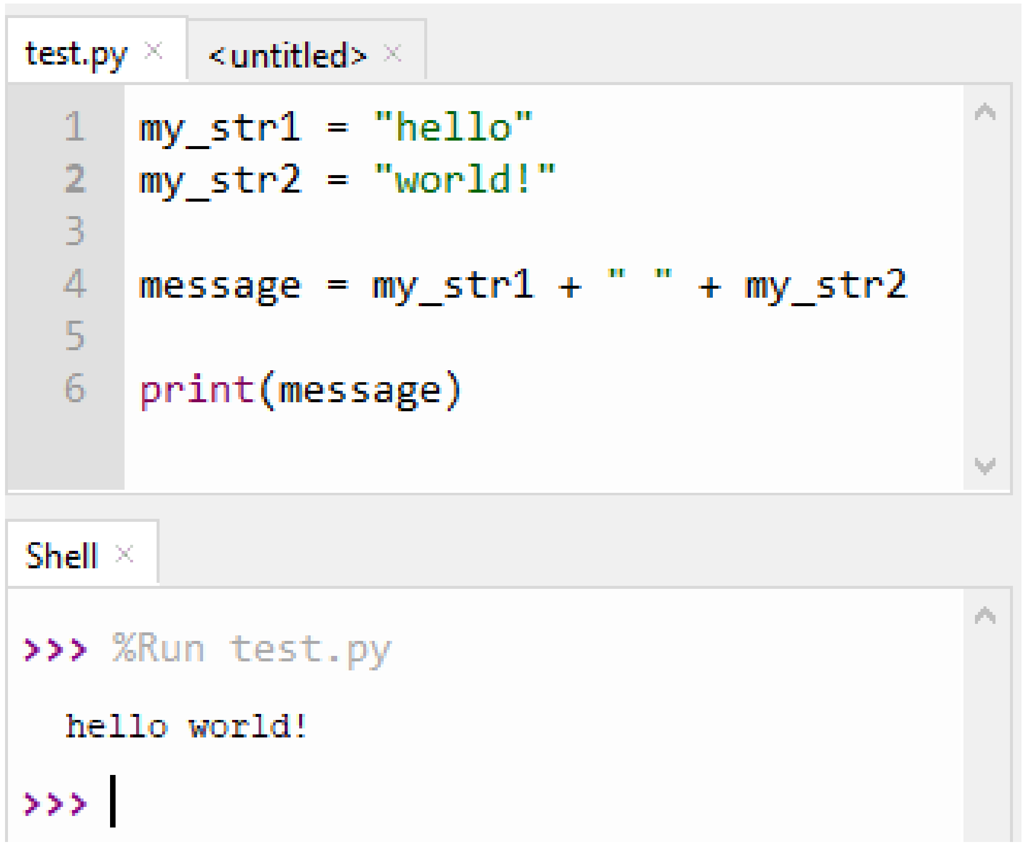


```python
my_int   = 1
my_float = 1.5

my_sum = my_int + my_float

print(my_sum)
print(type(my_sum))
```

```
>>> %Run test.py

  2.5
  <class 'float'>

>>>
```

Arithmetic

Some operators do special things



```
my_str1 = "hello"
my_str2 = "world!"

message = my_str1 + " " + my_str2

print(message)
```

```
>>> %Run test.py

  hello world!

>>> 
```

Logic

Operators

Equal	==
Not equal	!=
Greater than	>
Less than	<
Greater/equal	>=
Lesser/equal	<=

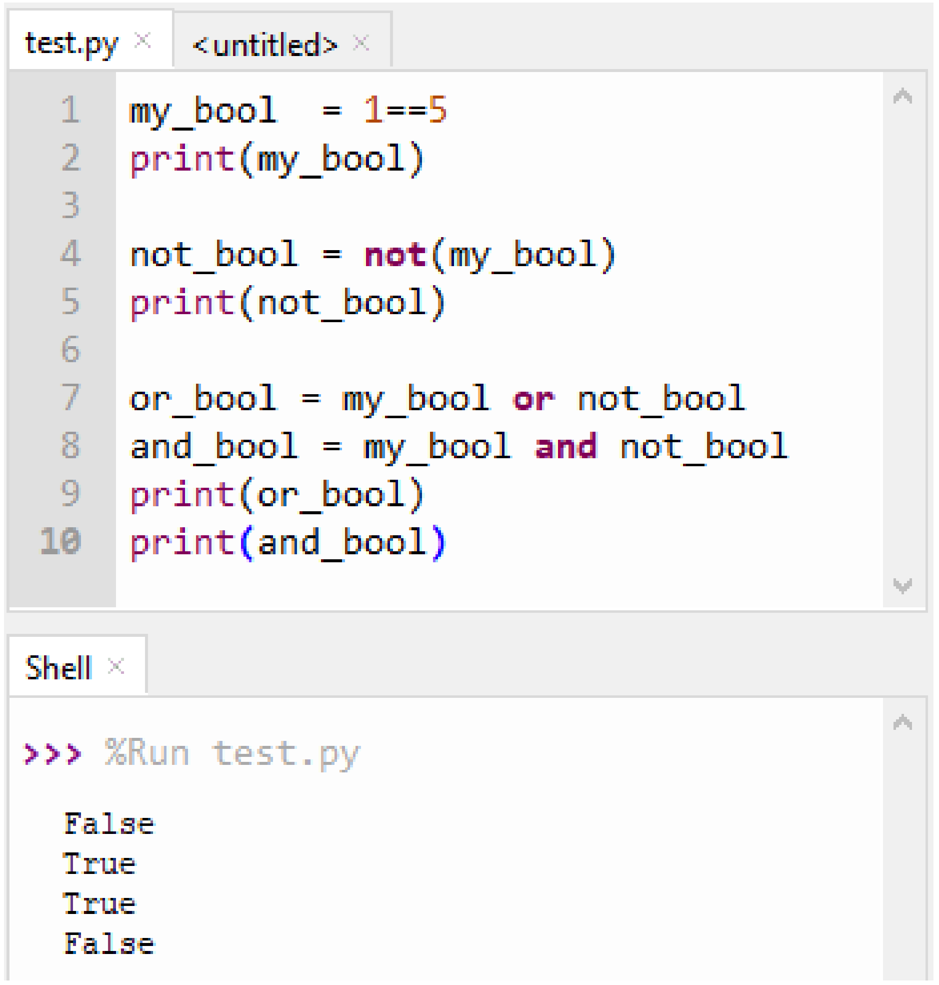
Logic

Conditions

And	X and Y
Or	X or Y
Not	not(X)

Logic

Example



test.py × <untitled> ×

```python
my_bool  = 1==5
print(my_bool)

not_bool = not(my_bool)
print(not_bool)

or_bool = my_bool or not_bool
and_bool = my_bool and not_bool
print(or_bool)
print(and_bool)
```

Shell ×

```
>>> %Run test.py

  False
  True
  True
  False
```

Next on #2

Functions, Methods and Packages

1

Python Quick Tips

Data Types