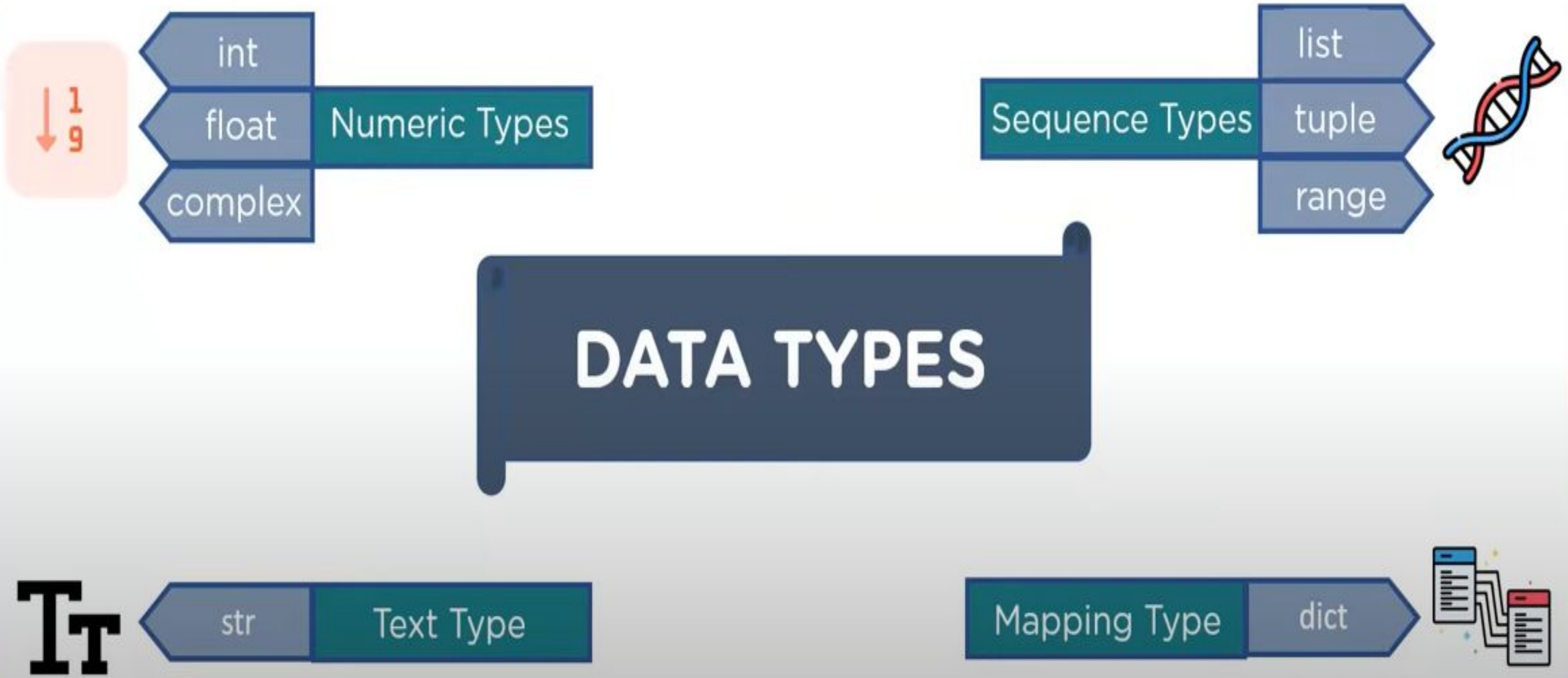
Python Data Types, Comments

What’s in it for you?

- What is a variable?
- Data types of variables
- Rules for naming variables
- Arithmetic operations with integer and float variables
- Operations on string variables
- Exercise on Strings

Data Types in Python

DATA TYPES



Numeric Types

- int
- float
- complex

Sequence Types

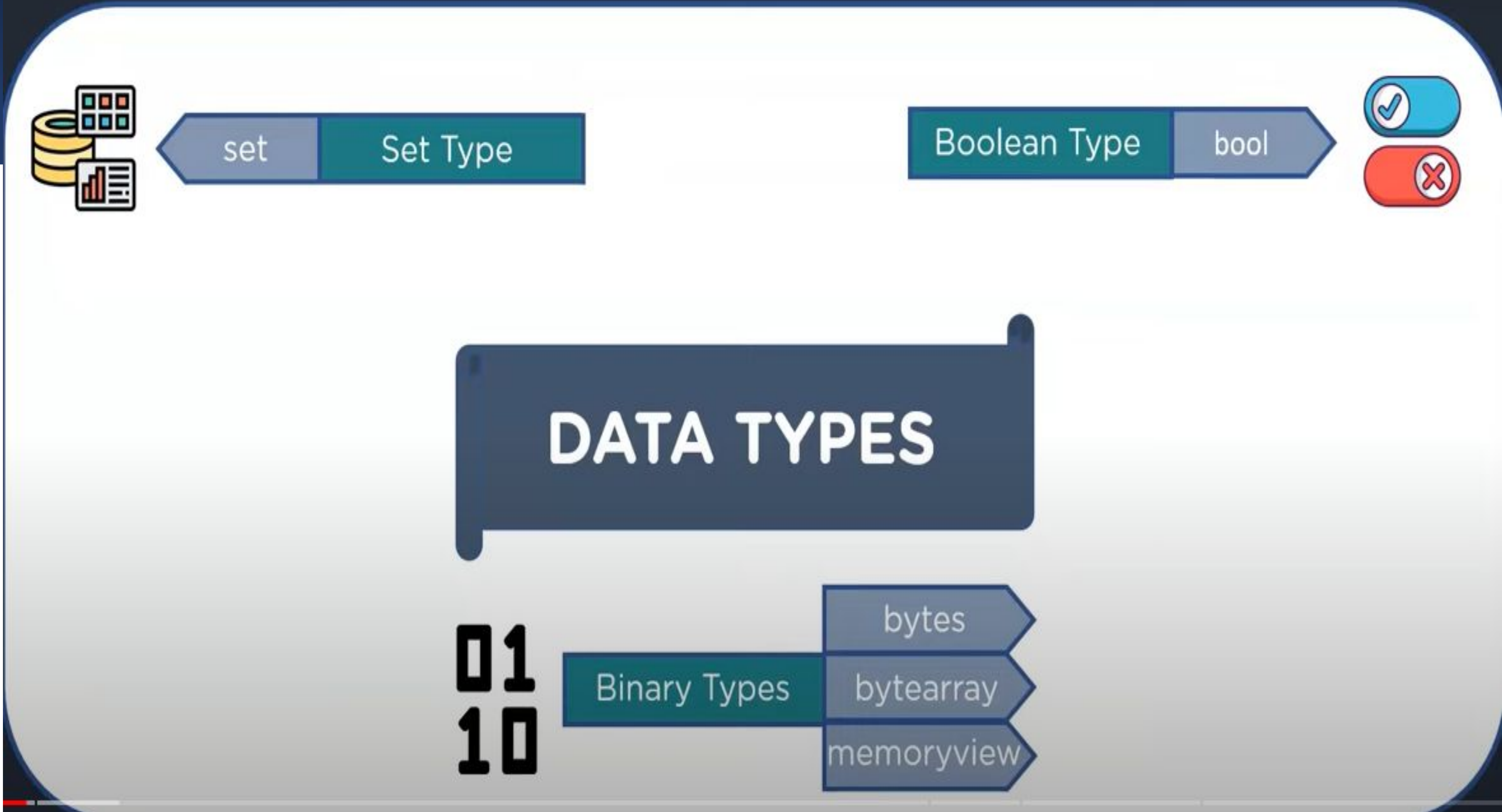
- list
- tuple
- range

Text Type

- str

Mapping Type

- dict



DATA TYPES

Set Type

set

Boolean Type

bool

Binary Types

bytes

bytearray

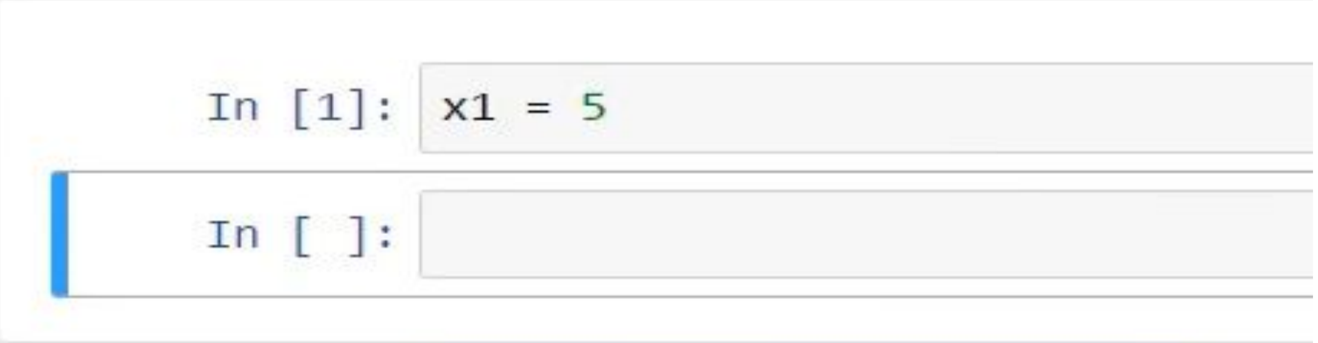
memoryview

Numeric

- int
- float
- complex

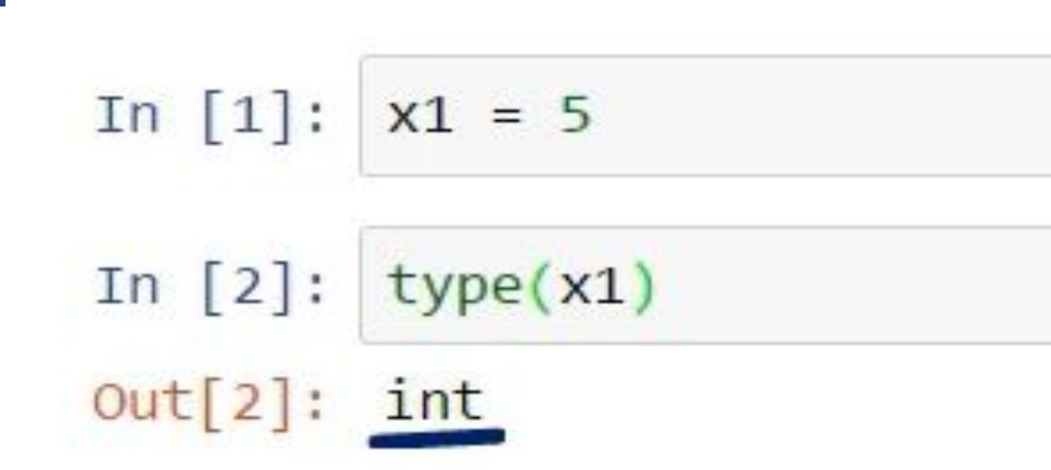
Integers:

Positive or negative whole numbers; without a decimal point

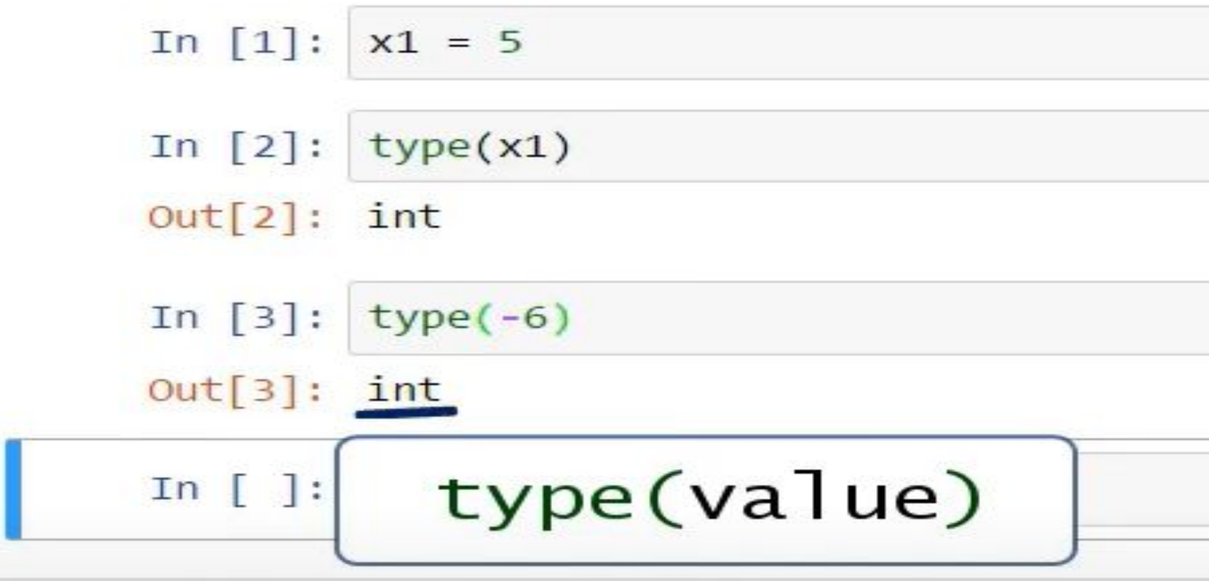


```
In [1]: x1 = 5

In [ ]:
```

type(var)

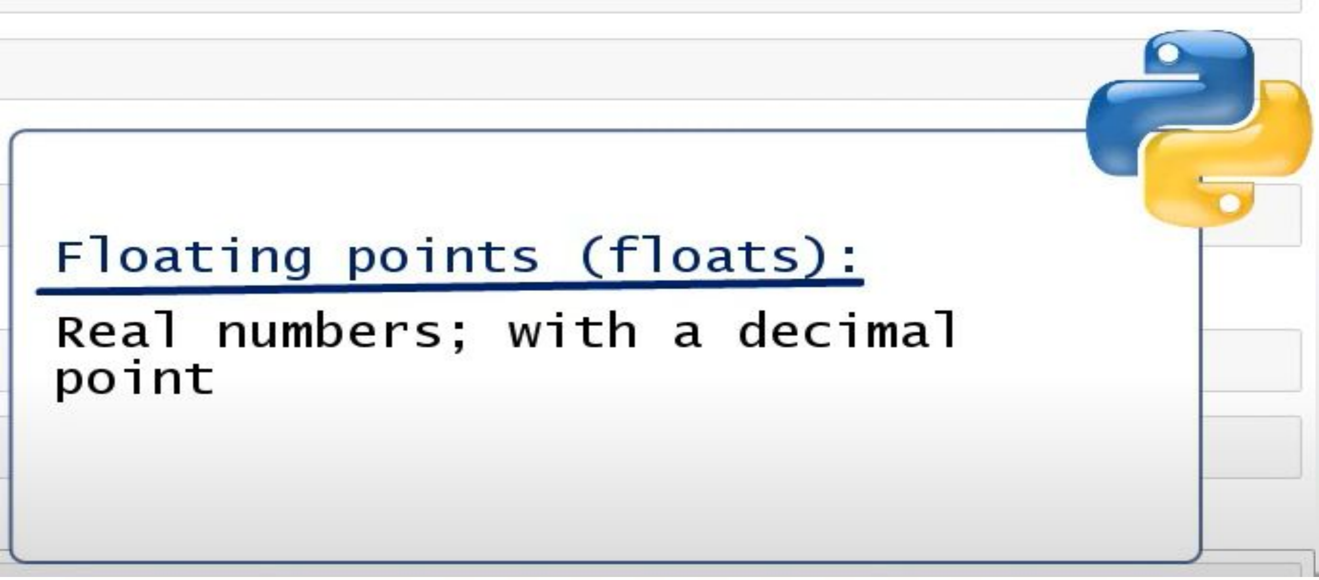


```
In [1]: x1 = 5

In [2]: type(x1)

Out[2]: int
```



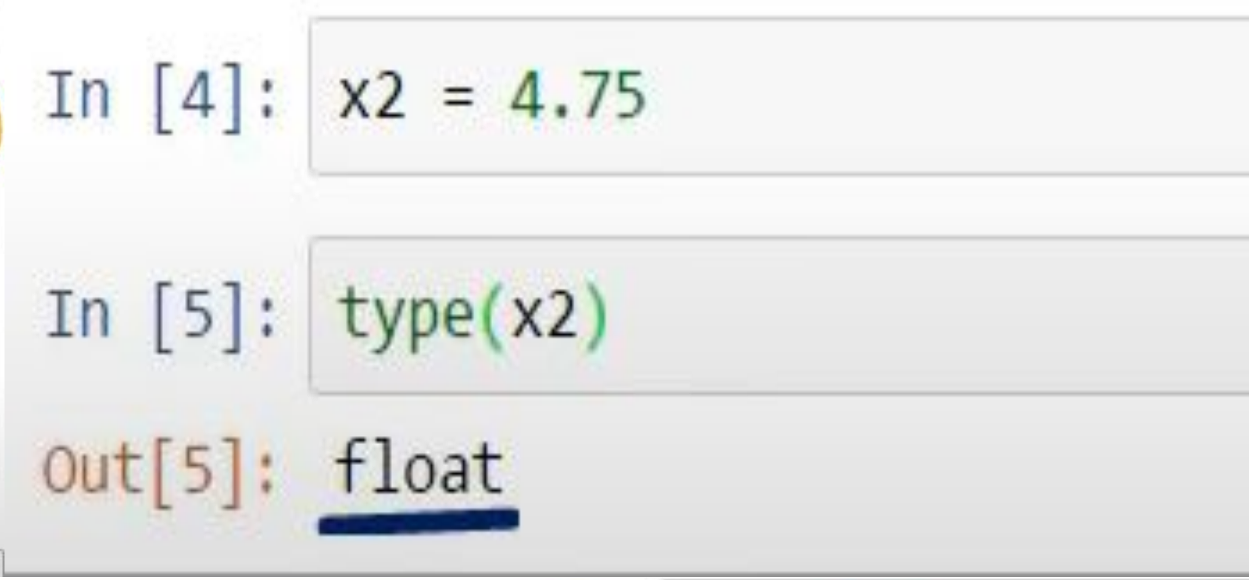
```
In [1]: x1 = 5

In [2]: type(x1)
Out[2]: int

In [3]: type(-6)
Out[3]: int

In [ ]:
```

type(value)

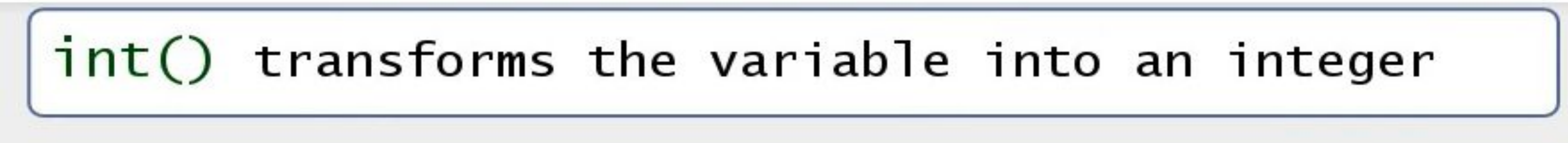


Floating points (floats):

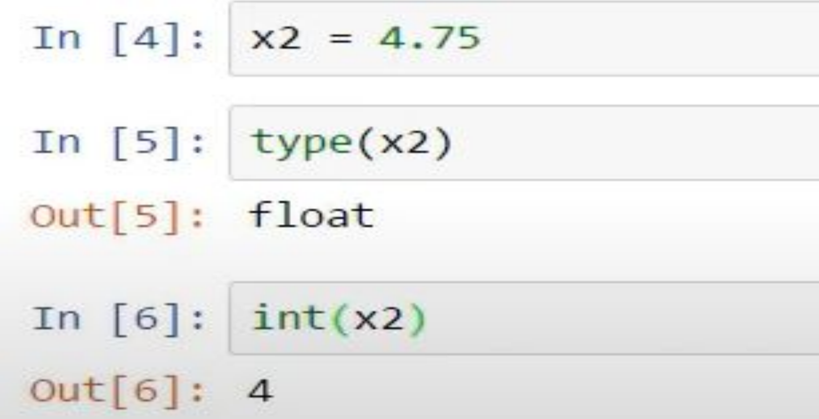
Real numbers; with a decimal point



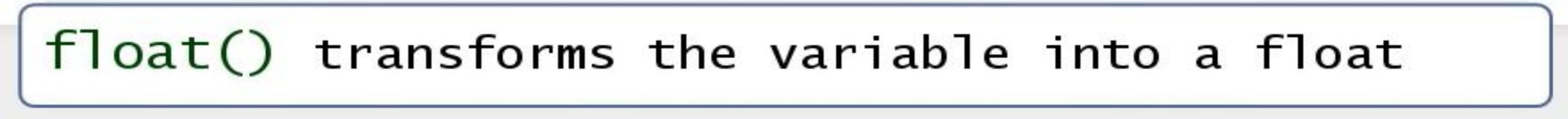
```
In [4]: x2 = 4.75

In [5]: type(x2)
Out[5]: float
```



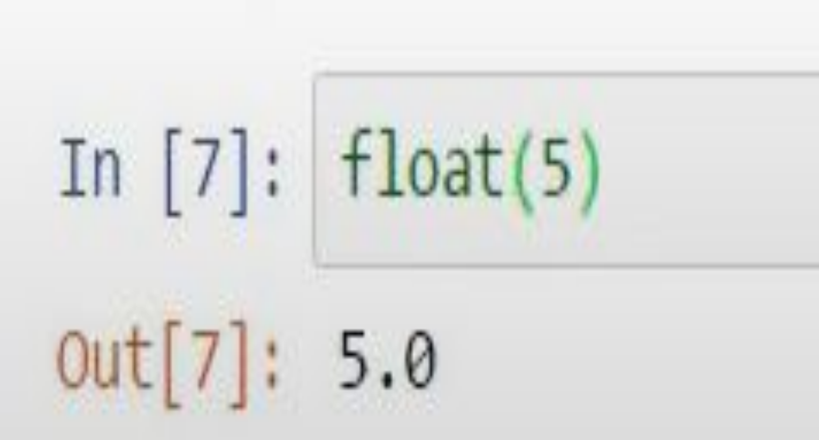
int() transforms the variable into an integer



```
In [4]: x2 = 4.75

In [5]: type(x2)
Out[5]: float

In [6]: int(x2)
Out[6]: 4
```



float() transforms the variable into a float



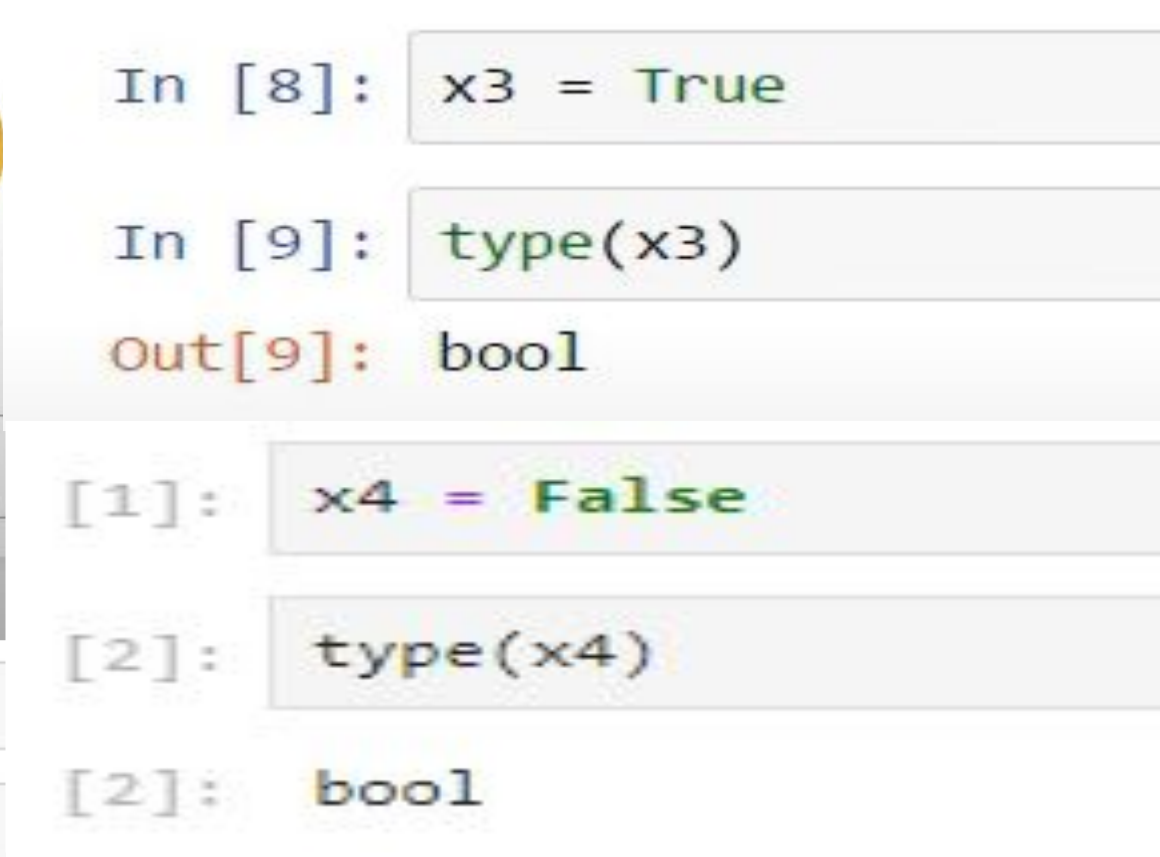
```
In [7]: float(5)

Out[7]: 5.0
```

Boolean values:

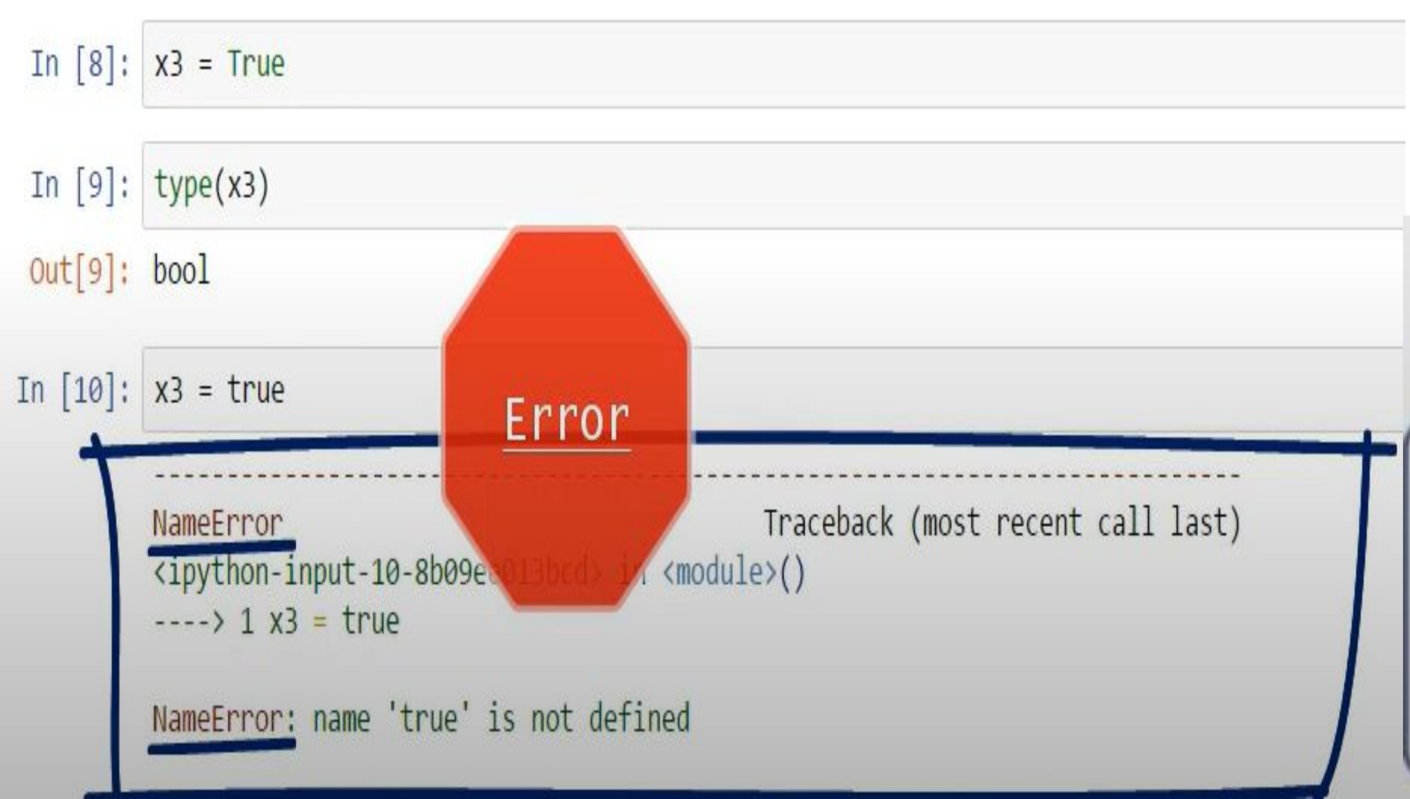
A "True" or "False" value

1 or 0

on or off

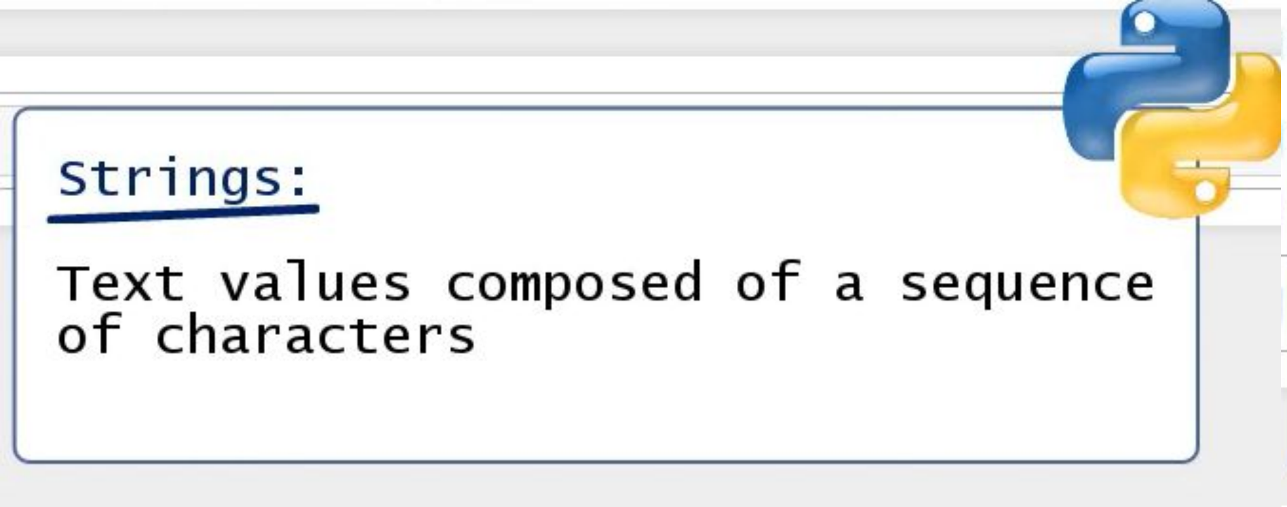


```
In [8]:  x3 = True

In [9]:  type(x3)
Out[9]:  bool

[1]:  x4 = False

[2]:  type(x4)

[2]:  bool
```



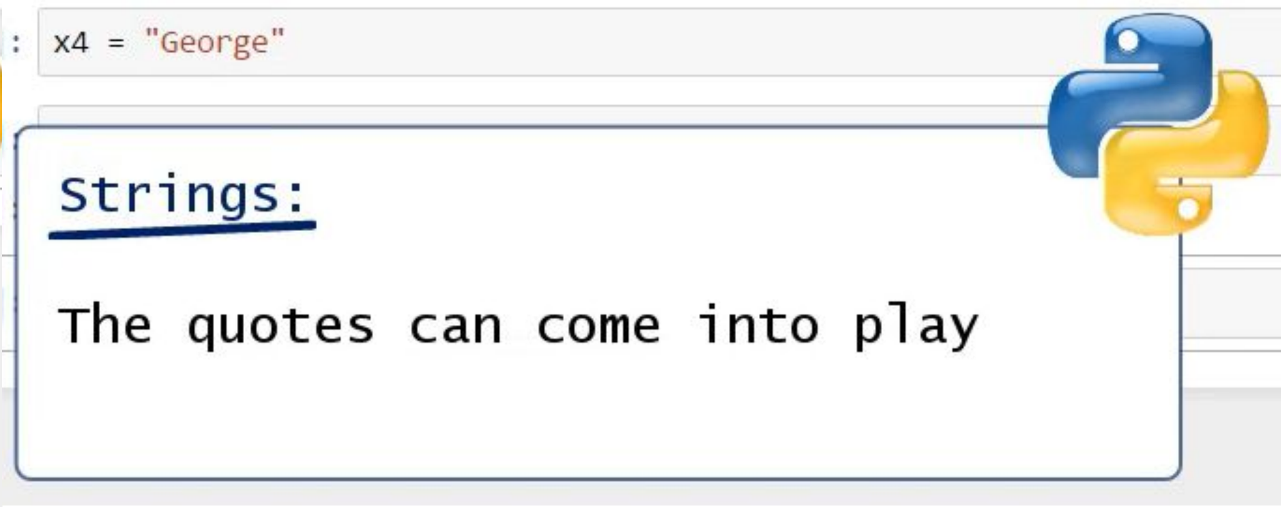
```
In [8]: x3 = True

In [9]: type(x3)
Out[9]: bool

In [10]: x3 = true
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-10-8b09e013bcd> in <module>()
----> 1 x3 = true

NameError: name 'true' is not defined
```

True; False



Strings:

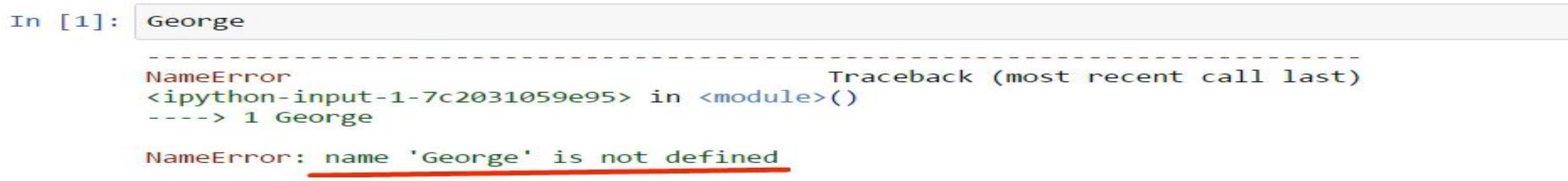
Text values composed of a sequence of characters



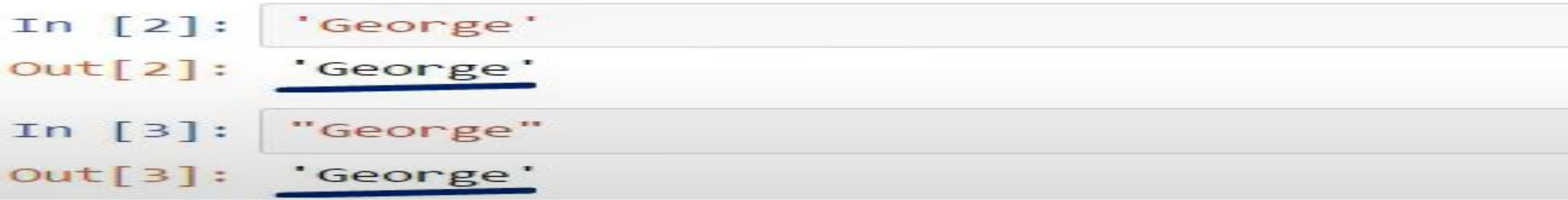
```
x4 = "George"
```

Strings:

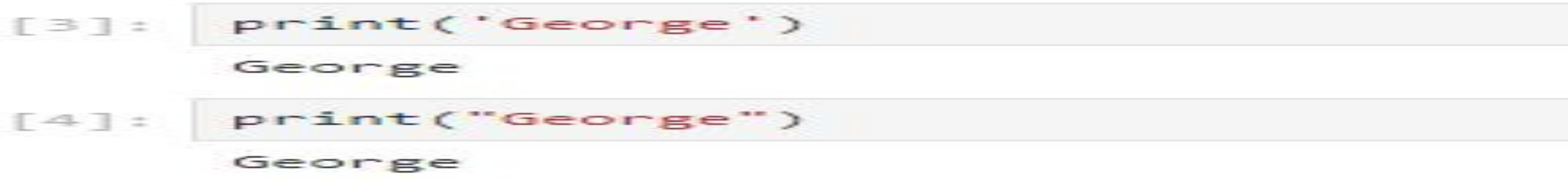
The quotes can come into play



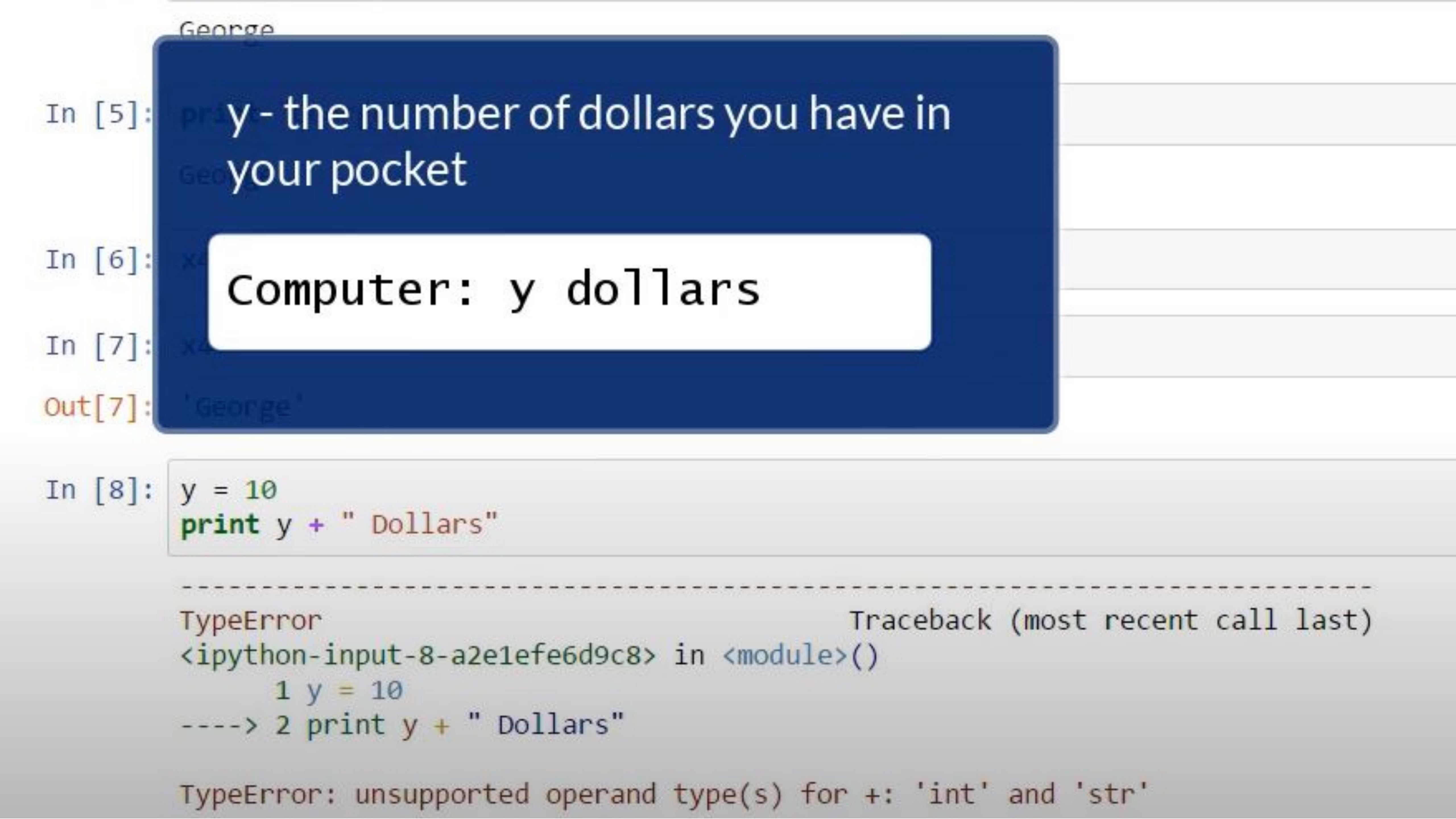
```
In [1]: George

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-7c2031059e95> in <module>()
----> 1 George

NameError: name 'George' is not defined
```



```
In [2]: 'George'
Out[2]: 'George'

In [3]: "George"
Out[3]: 'George'
```



```
[3]: print('George')
     George

[4]: print("George")
     George
```

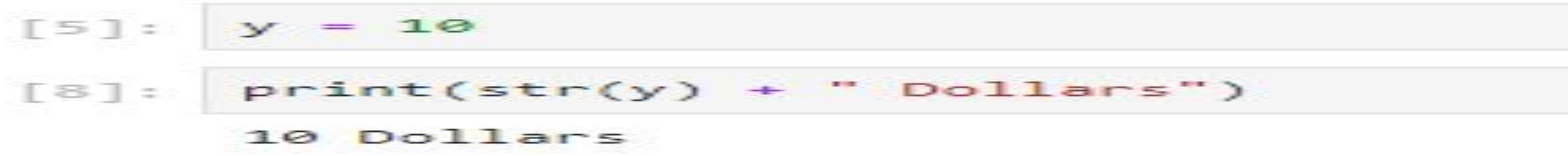


y - the number of dollars you have in your pocket

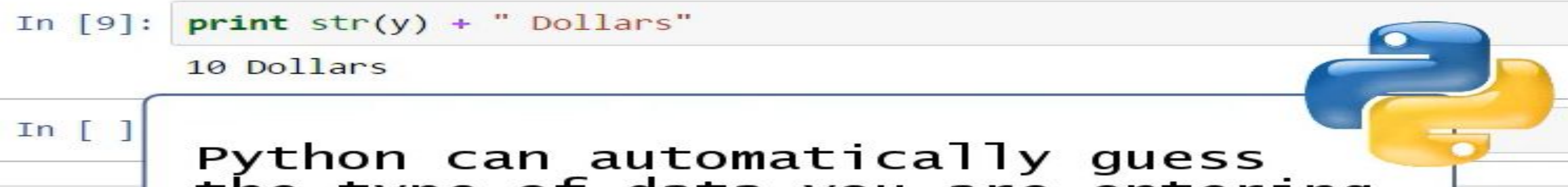
```
Computer: y dollars
```

```
George

In [5]: pri

        Geo

In [6]: x4

In [7]: x4

Out[7]: 'George'

In [8]: y = 10
        print y + " Dollars"

        ---------------------------------------------------------------------------
        TypeError                                 Traceback (most recent call last)
        <ipython-input-8-a2e1efe6d9c8> in <module>()
              1 y = 10
        ----> 2 print y + " Dollars"

        TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

str() converts a number into text

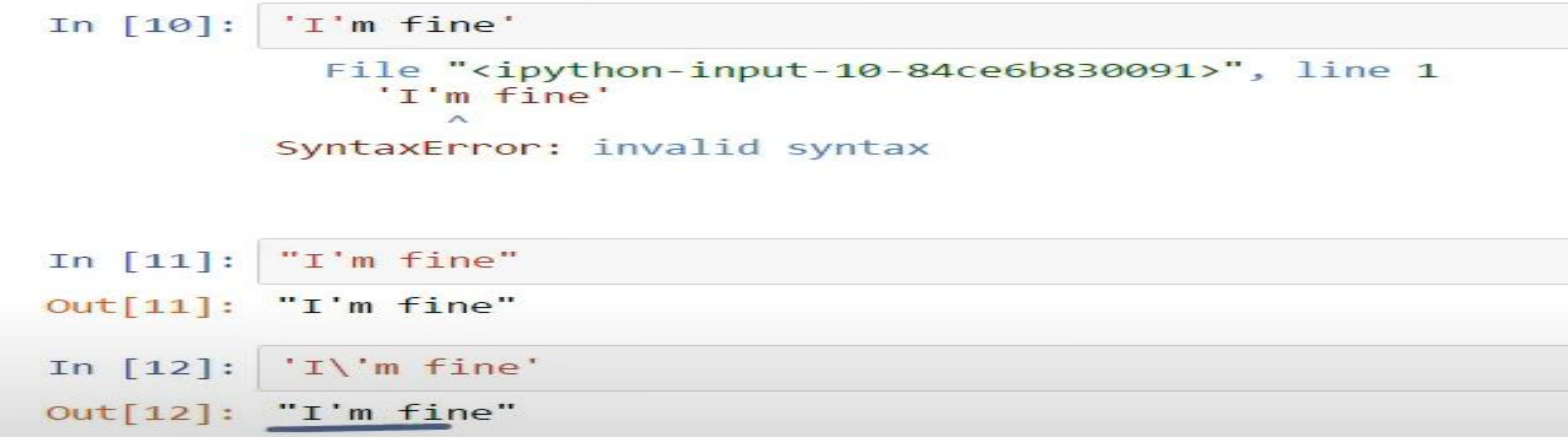


```
[5]: y = 10

[8]: print(str(y) + " Dollars")

     10 Dollars
```



```
In [9]: print str(y) + " Dollars"
        10 Dollars

In [ ]
```

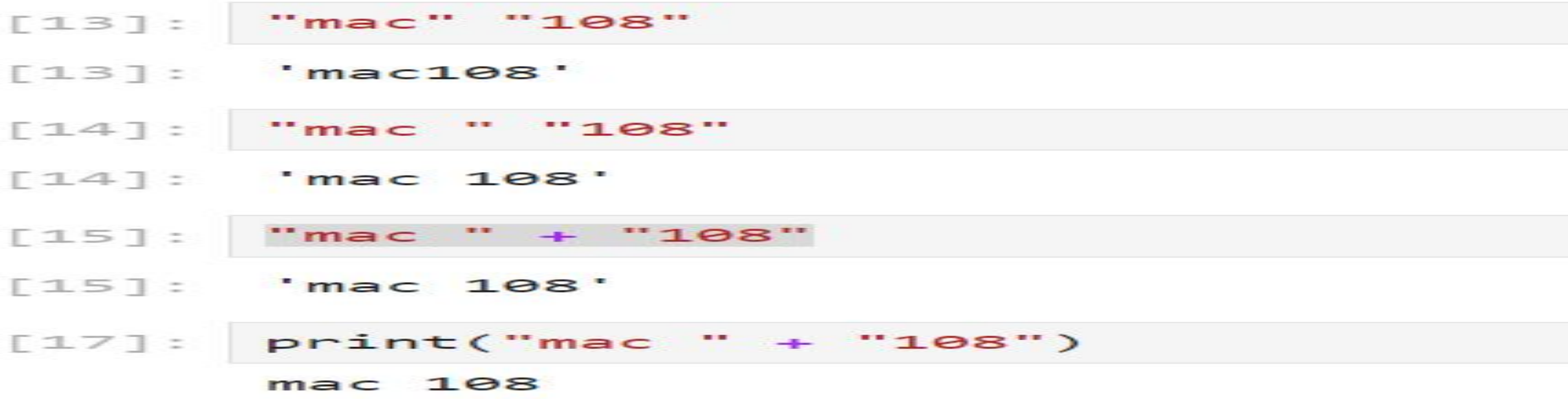
Python can automatically guess the type of data you are entering



```
In [10]: 'I'm fine'

           File "<ipython-input-10-84ce6b830091>", line 1
             'I'm fine'
                ^
         SyntaxError: invalid syntax


In [11]: "I'm fine"
Out[11]: "I'm fine"

In [12]: 'I\'m fine'
Out[12]: "I'm fine"
```

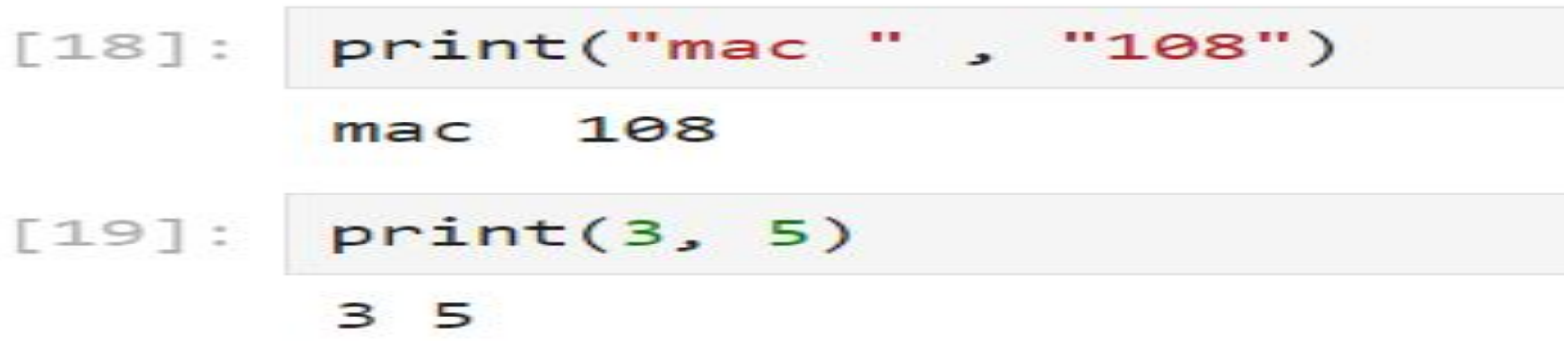
```
In [ ]
```

\ escape character



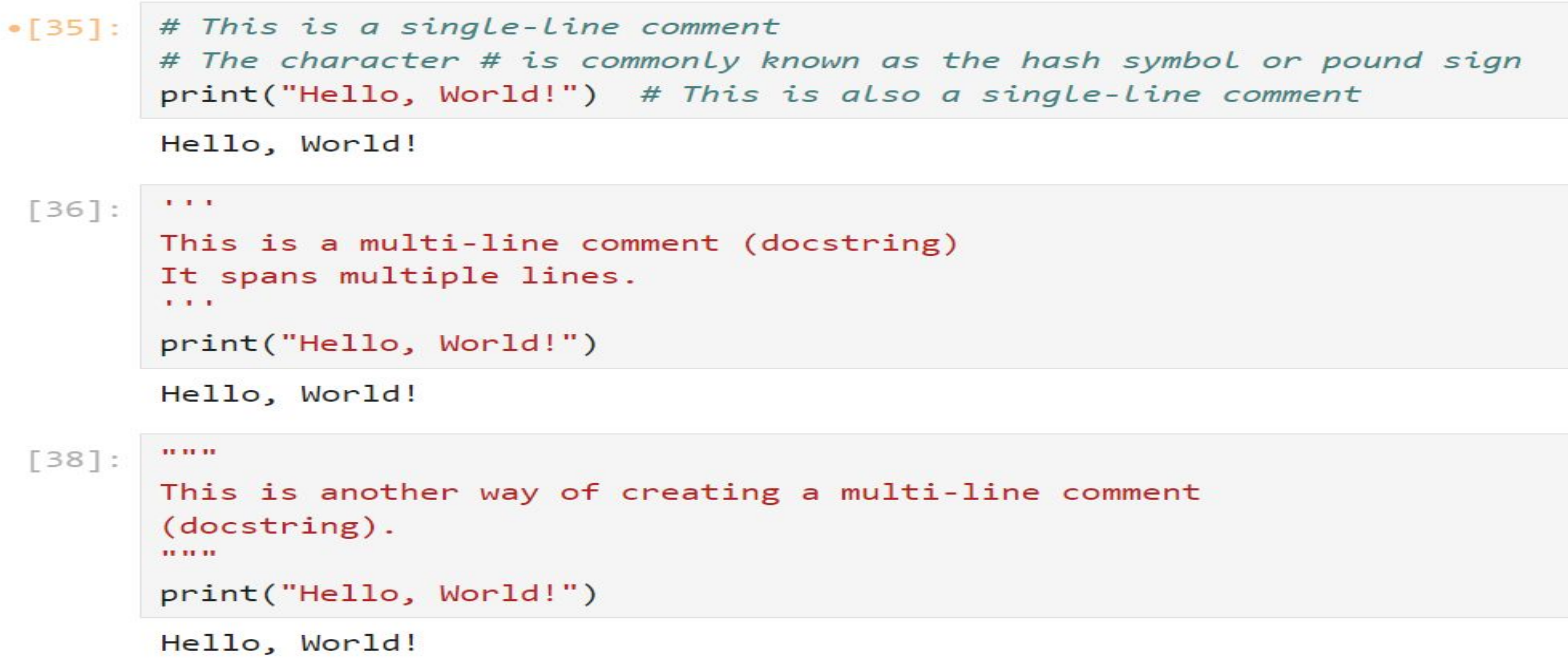
```
[13]: "mac" "108"
[13]: 'mac108'
[14]: "mac " "108"
[14]: 'mac 108'
[15]: "mac " + "108"
[15]: 'mac 108'
[17]: print("mac " + "108")
      mac 108
```

, trailing comma



```
[18]: print("mac " , "108")
      mac  108
[19]: print(3, 5)
      3 5
```

Python Comments



•[35]:

```python
# This is a single-line comment
# The character # is commonly known as the hash symbol or pound sign
print("Hello, World!")  # This is also a single-line comment
```

```
Hello, World!
```

[36]:

```python
'''
This is a multi-line comment (docstring)
It spans multiple lines.
'''
print("Hello, World!")
```

```
Hello, World!
```

[38]:

```python
"""
This is another way of creating a multi-line comment
(docstring).
"""
print("Hello, World!")
```

```
Hello, World!
```