Python

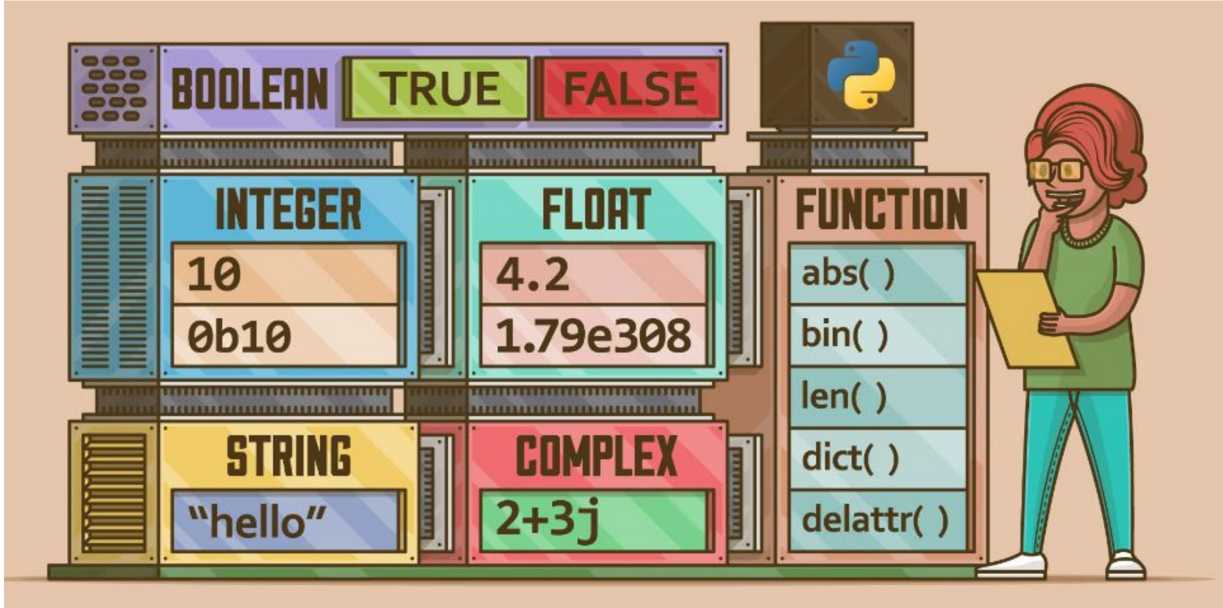
Lesson 5

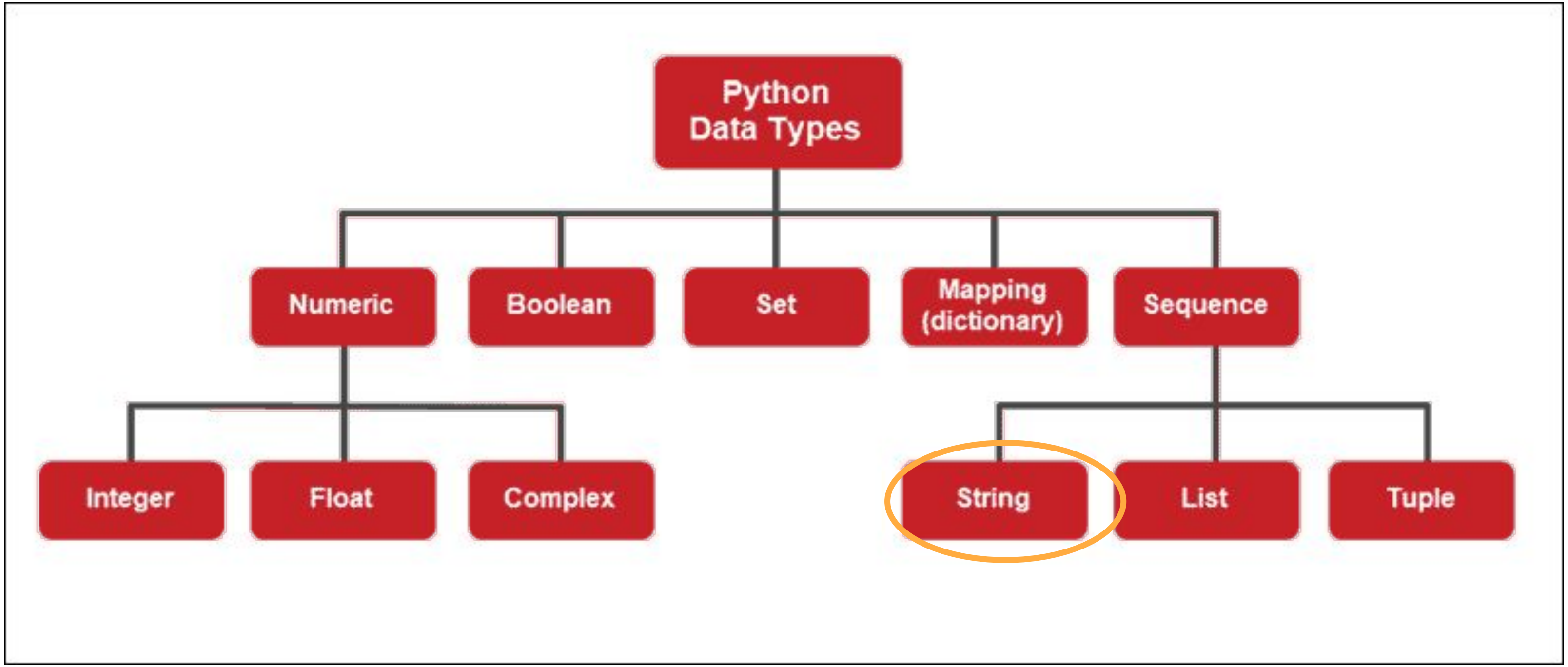
Strings, Lists, Tuples, and Maps

Data types

Strings, lists, tuples, and maps are all different types of data structures in Python, and they have different properties and uses.

(remember Integer?)





Python Data Types

Numeric

Boolean

Set

Mapping (dictionary)

Sequence

Integer

Float

Complex

String

List

Tuple

String

In programming terms, we usually call text a string. When you think of a string as a collection of letters, the term makes sense.

All the letters, numbers, and symbols could be a string.

Creating Strings

In Python, we create a string by putting quotes around text.

```
fred = "Why do gorillas have big nostrils? Big fingers!!"
```

You can also use **single quotes** to create a string, like this:

```
fred = 'What is pink and fluffy? Pink fluff!!'
```

Handling Problems with Strings

Now consider this crazy example of a string, which causes Python to display an error message:

```
silly_string = 'He said, "Aren't can't shouldn't wouldn't."'
```

SyntaxError: invalid syntax

In the first line, we try to create a string (defined as the variable silly_string) enclosed by single quotes, but also containing a mixture of single quotes in the words can't, shouldn't, and wouldn't, as well as double quotes. What a mess!

The solution to this problem is a multiline string using three single quotes ('''), which allows us to combine double and single quotes in our string without causing errors.

Three single quotes

If we use three single quotes, we can put any combination of single and double quotes inside the string:

```
silly_string = '''He said, "Aren't can't shouldn't wouldn't."'''
```

Backslash (\) before each quotation mark

You can add a backslash (\) before each quotation mark within the string. This is called ***escaping***. It’s a way of saying to Python, “Yes, I know I have quotes inside my string, and I want you to ignore them until you see the end quote.”

```
single_quote_str = 'He said, "Aren\'t can\'t shouldn\'t wouldn\'t."'
```

create a string with single quotes, using the backslash in front of the single quotes inside that string.

```
double_quote_str = "He said, \"Aren't can't shouldn't wouldn't.\""
```

create a string with double quotes, and use the backslash in front of those quotes in the string.

Embedding values is programmer-speak for “inserting values.”

Embedding Values in Strings

```
myscore = 1000

message = 'I scored %s points'

print (message % myscore)
```

If you want to display a message using the contents of a variable, you can embed values in a string using ***%s***

use %s in the sentence in place of the value

Question

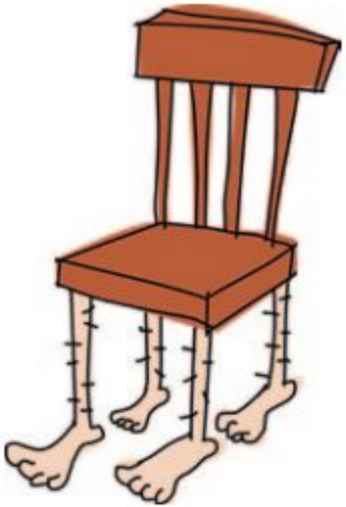
```
joke_text = '%s: a device for finding furniture in the dark'

bodypart1 = 'Knee'

bodypart2 = 'Shin'

print(joke_text % bodypart1)

print(joke_text % bodypart2)
```



What should be the expected result of the coding above?

Question

```
nums = 'What did the number %s say to the number %s? Nice belt!!'

print(nums % (0, 8))
```

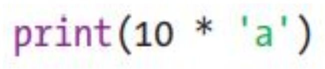
What should be the expected result of the coding above?

When using more than one placeholder, be sure to wrap the replacement values in parentheses, as shown in the example. The order of the values is the order in which they’ll be used in the string.

Multiplying Strings

What is 10 multiplied by 5? The answer is 50, of course.

But what’s 10 multiplied by a?



```
print(10 * 'a')
```

Space

Try to print out a string like this:

There are 20 spaces between a   and b.

Homework 1 - 10 minutes video time

Print out the silly message:

Good morning,

This is Austin. I’ll 20 eat donuts today..........aha!

Requirements:

- Using single quote mark to create the string.
- There’s a blank line between the first and second line.
- Replace 20 with a variable ‘num_of_donut’
- There are 30 dots after ‘today’. Using a string multiply.