IOTA ACADEMY

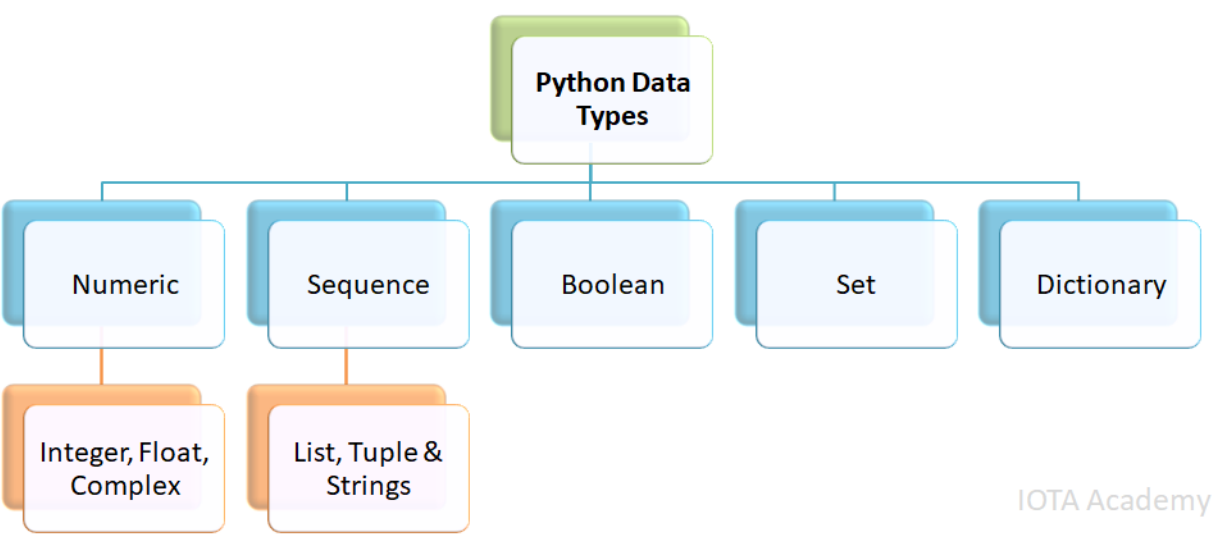
Become Job Ready...

Python Data Types

- Data types are the classification or categorization of data items.
- It represents the kind of value that tells what operations can be performed on a particular data.
- Since everything is an object in Python programming, data types are actually classes and variables are instance (object) of these classes.

Standard or built-in data type of Python

- Numeric
  > int, float and complex
- Sequence
  > string, list and tuple
- Boolean
- Set
- Dictionary



Numeric

In Python, numeric data type represent the data which has numeric value.

Numeric value can be integer, floating number or even complex numbers.

> These values are defined as **int, float and complex class in Python.**

Integer

This value is represented by **int class**.

It contains positive or negative whole numbers (without fraction or decimal).

In Python there is no limit to how long an integer value can be.

In [1]:
```
# Integer number

x = 10

print(x)
print(type(x))
```

```
10
<class 'int'>
```

Float

This value is represented by **float class**.

It is a real number with floating point representation.

It is specified by a decimal point.

Optionally, the character e or E followed by a positive or negative integer may be appended to specify scientific notation.

In [2]:
```
# float number

x=10.0

print(x)
print(type(x))
```

```
10.0
<class 'float'>
```

Complex

Complex number is represented by **complex** class.

It is specified as **(real part) + (imaginary part)j**. For example – 2+3j

In [3]:
```
# complex number

x = 4 + 5j  # real_part + imaginary_part j

print(x)
print(type(x))
```

```
(4+5j)
<class 'complex'>
```

Sequence

In Python, sequence is the **ordered collection of similar or different data types.**

Sequences allows to store multiple values in an organized and efficient fashion.

There are sequence types in Python –

- String
- List
- Tuple

Strings

A string is a collection of one or more characters put in a single quote, double-quote or triple quote.

In python there is no character data type.

It is represented by **str class**.

In [4]:
```
# creating string

#method 1: single quotes
x = 'I am a string'

#method 2: double quotes
y = "I am string in double quotes"

#method 3: triple quotes to write multiline string
z = """ I am string created using triple quotes.
I am another line of the same string."""

print(x)
print(type(x))

print(y)
print(type(y))

print(z)
print(type(z))
```

```
I am a string
<class 'str'>
I am string in double quotes
<class 'str'>
 I am string created using triple quotes.
I am another line of the same string.
<class 'str'>
```

Mutable vs Immutable

Mutable

Mutable objects can change their state or contents.

These are of in-built types like **list, dictionary, set.**
In simple words, a mutable object **can be changed** after it is created.

Immutable

Immutable objects can't change their state or content.

These are of in-built types like **int, float, bool, string, unicode, tuple.**
In simple words, an immutable object **can not be changed** after it is created.

That's Great!