python™

Alexander Schober

Summary

09:00 - 10:00 Installing Python and BornAgain

10:00 - 10:05 What is Python

10:05 - 10:40 Python basics

10:40 - 11:00 Coffee break

11:00 - 11:30 NumPy introduction

11:30 - 12:00 NumPy advanced

12:00 - 12:30 Matplotlib

12:30 - 13:00 Advanced python

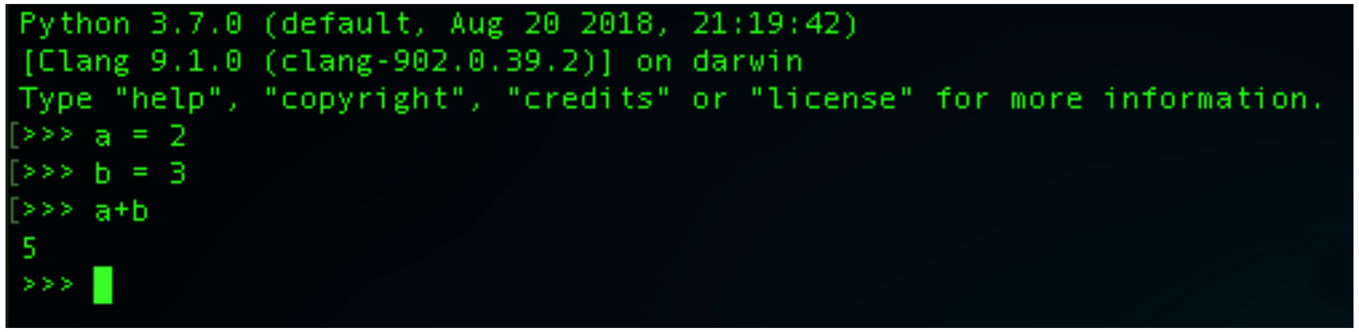
What is Python ?

Python is an interpreted, high-level programming language for general-purpose programming

Interpreted language

Translates code into machine language without previous compilation

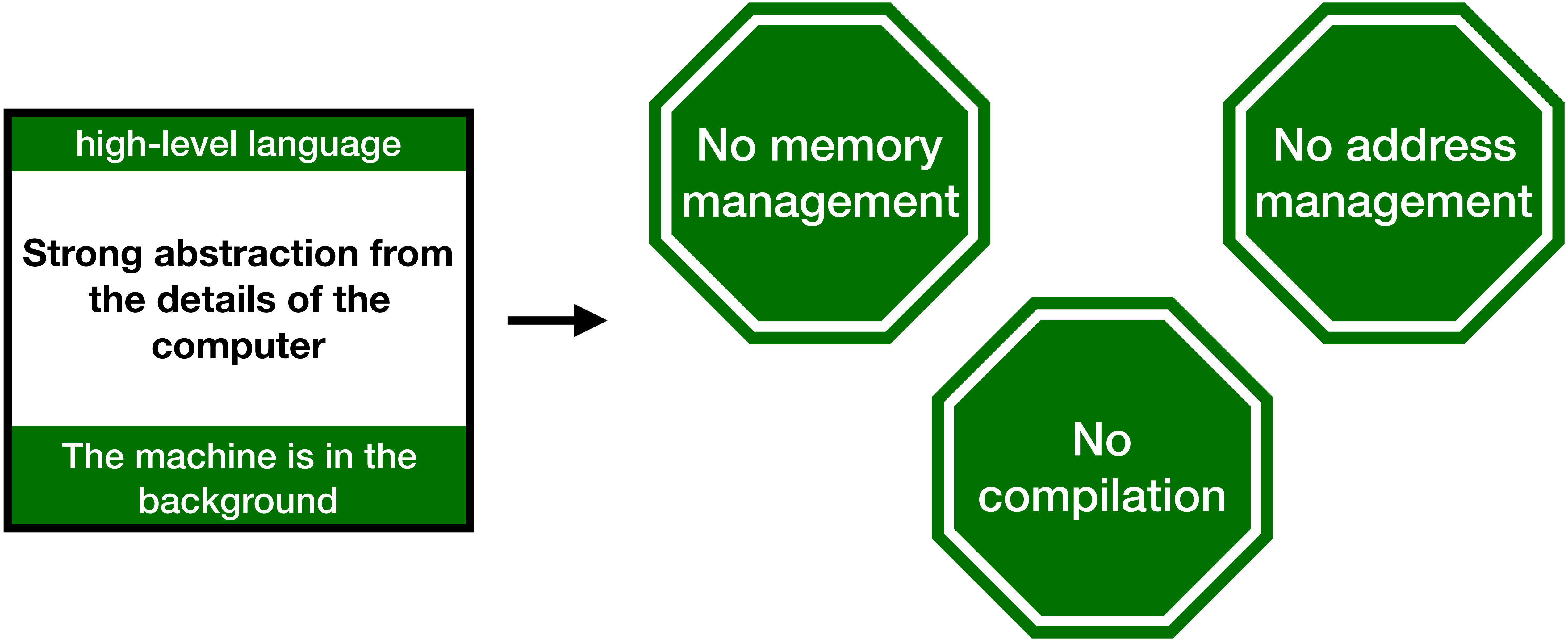
Write and run …



```
Python 3.7.0 (default, Aug 20 2018, 21:19:42)
[Clang 9.1.0 (clang-902.0.39.2)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> a = 2
[>>> b = 3
[>>> a+b
5
>>> 
```

What is Python ?

Python is an interpreted, high-level programming language for general-purpose programming



high-level language

Strong abstraction from the details of the computer

The machine is in the background

No memory management

No address management

No compilation

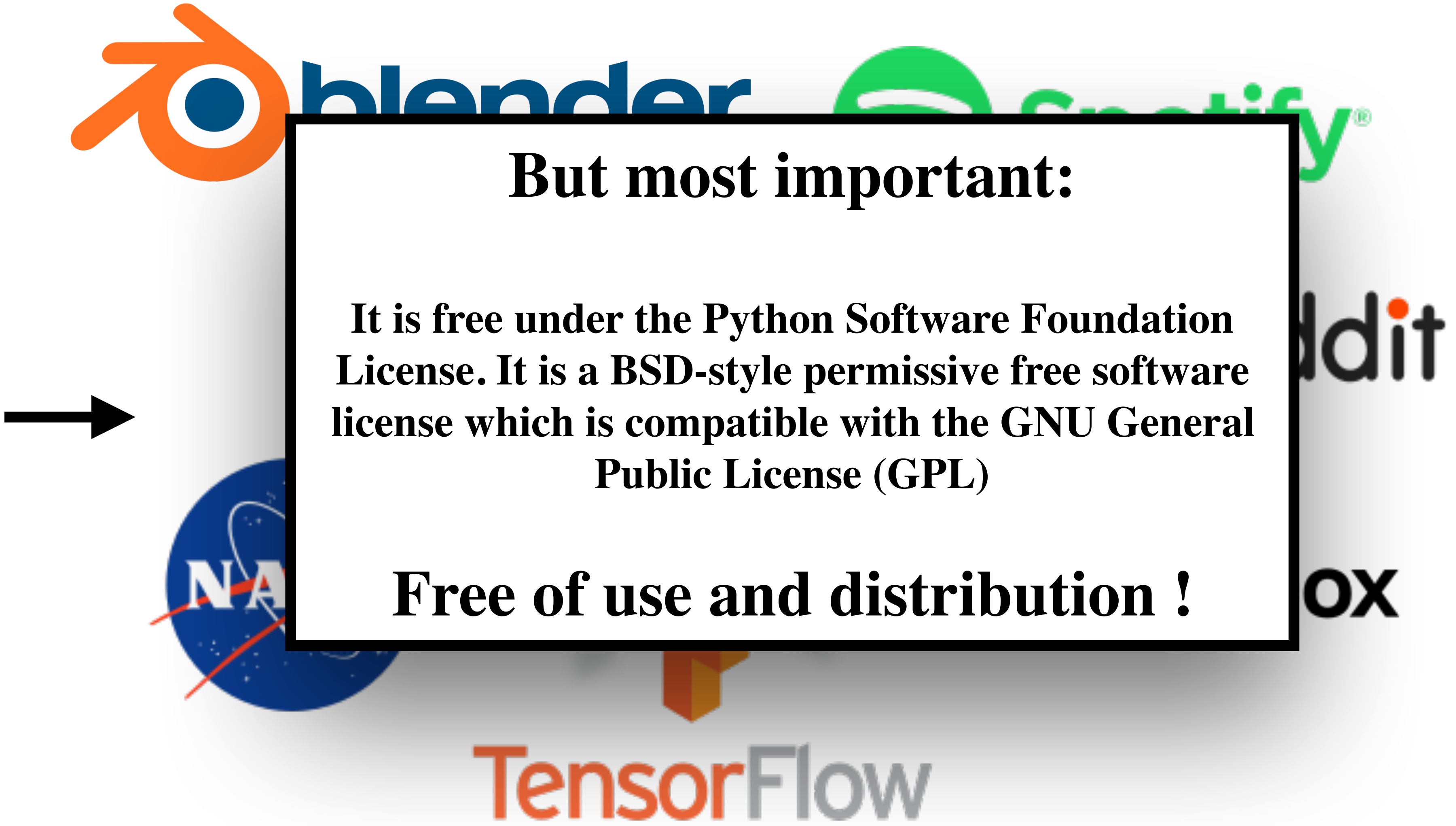
What is Python ?

Python is an interpreted, high-level programming language for general-purpose programming

general-purpose programming

Versatile !
Not designed to solve one problem in particular…

→

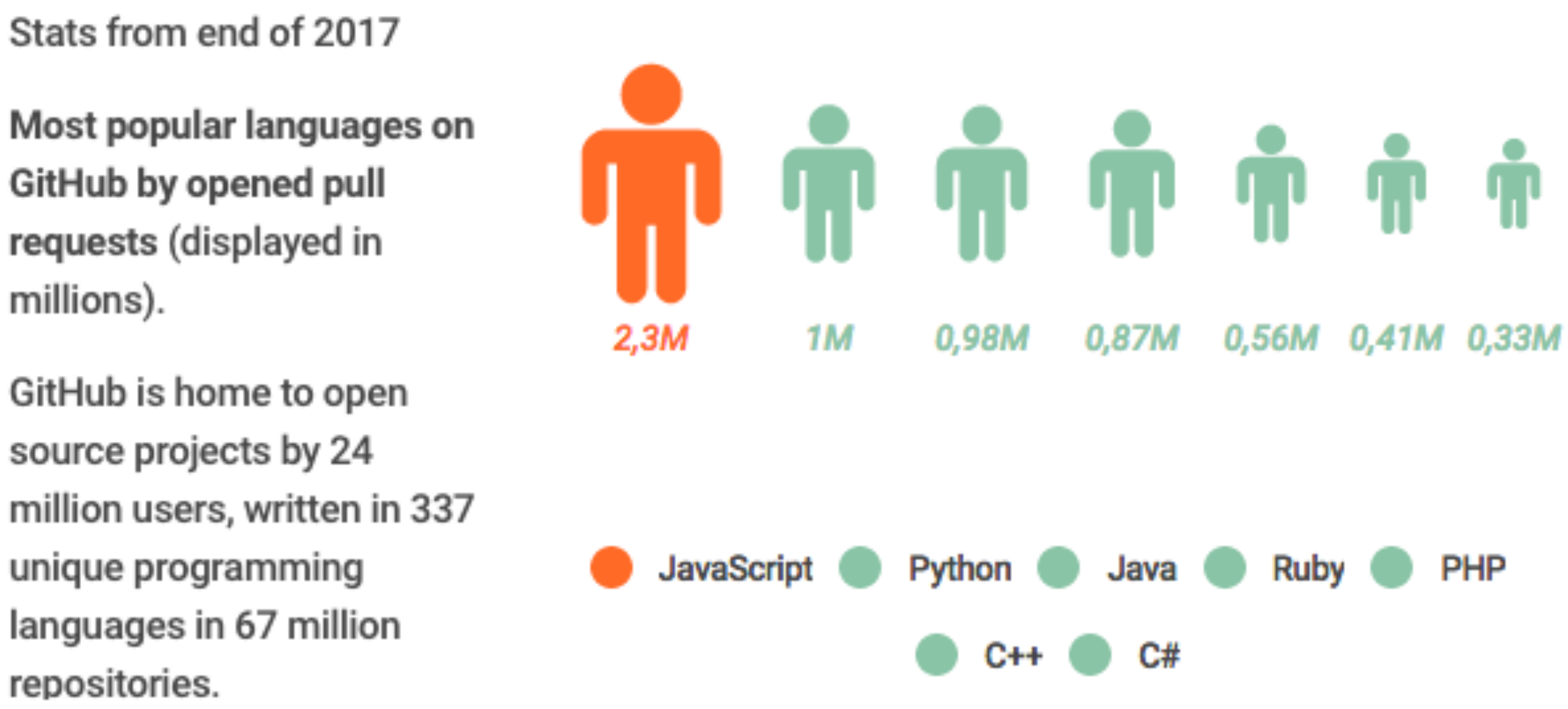


But most important:

It is free under the Python Software Foundation License. It is a BSD-style permissive free software license which is compatible with the GNU General Public License (GPL)

Free of use and distribution !

What is Python ?



Stats from end of 2017

Most popular languages on GitHub by opened pull requests (displayed in millions).

GitHub is home to open source projects by 24 million users, written in 337 unique programming languages in 67 million repositories.

Let's add to this number …

Summary

09:00 - 10:00 Installing Python and BornAgain

10:00 - 10:05 What is Python

10:05 - 10:40 Python basics

10:40 - 11:00 Coffee break

11:00 - 11:30 NumPy introduction

11:30 - 12:00 NumPy advanced

12:00 - 12:30 Matplotlib

12:30 - 13:00 Advanced python

Python Basics

Open a notebook …

Python Builtin Types

- Numeric datatypes:

  Integers: *a = 2*

  Float: *a = 2.0*

  Complex: *a = 1.0j*

- Strings: *a = ‘hello'* or *a = “hello”* or *a = '''hello'''*
- Booleans: *a* = *True* or *a* = *False*
- Sequences:

  Lists: a = [0, 1, 2, 3] (mutable)

  Tuples: a = (0, 1, 2, 3) (immutable)

  Sets: a = {0, 1, 2, 3} (mutable)

- Dictionaries: a = {‘house’:’red’, ‘garden':'green' }

Python Builtin Operands

Numeric datatypes

- Addition: 3 + 2
- Subtraction: 3 - 2
- Multiplication: 3 * 2
- Exponentiation: 3 ** 2
- Division: 3 / 2
- Modulus: 3 % 2

Strings

Addition: “hello" + “bye"

Lists

Addition: [1,2,3] + [4,5,6]

Tuples

Addition: (1,2,3) + (4,5,6)

Python Boolean logic

Comparison:

Boolean types take the values True or False. The result of a comparison operator is boolean.

Smaller then:

5 < 6 ➤ True

Bigger or equal then:

5 >= 6 ➤ False

Equal to:

5 == 6 ➤ False

Logical operations:

It is also possible to perform some logical operations to prove certain statements:

and:

5 >= 6 and 5 < 6 ➤ False

or:

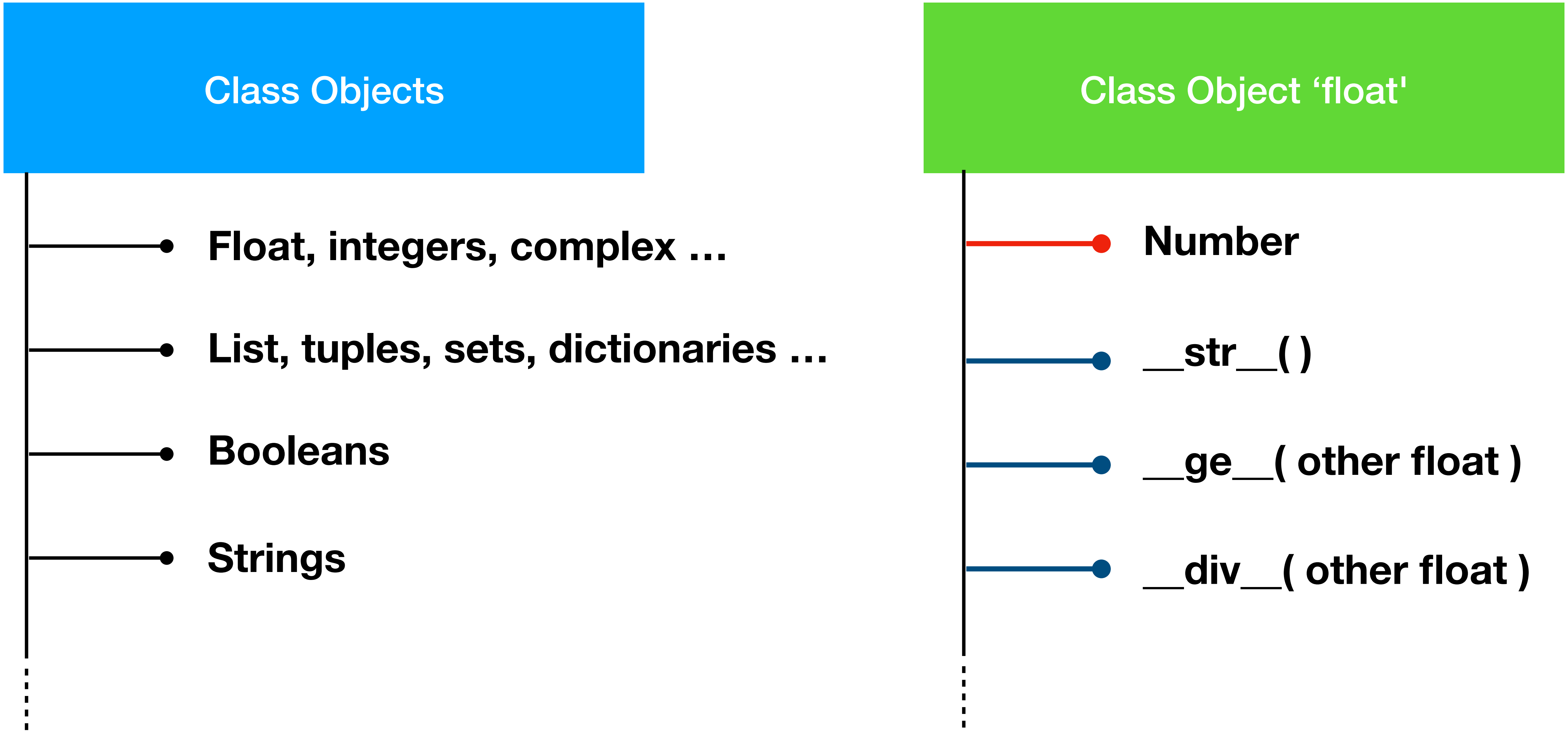
5 >= 6 or 5 < 6 ➤ True

not:

not 5 == 6 ➤ True

Python Objects

Everything defined within python is an object:



Class Objects

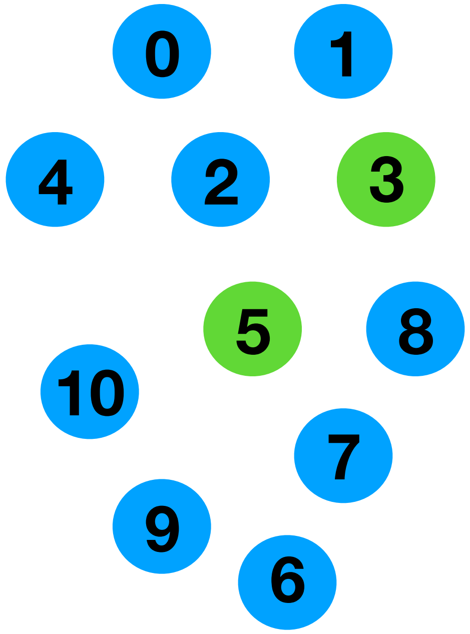
- Float, integers, complex …
- List, tuples, sets, dictionaries …
- Booleans
- Strings

Class Object ‘float'

- Number
- __str__( )
- __ge__(other float)
- __div__(other float)

Python Sequences

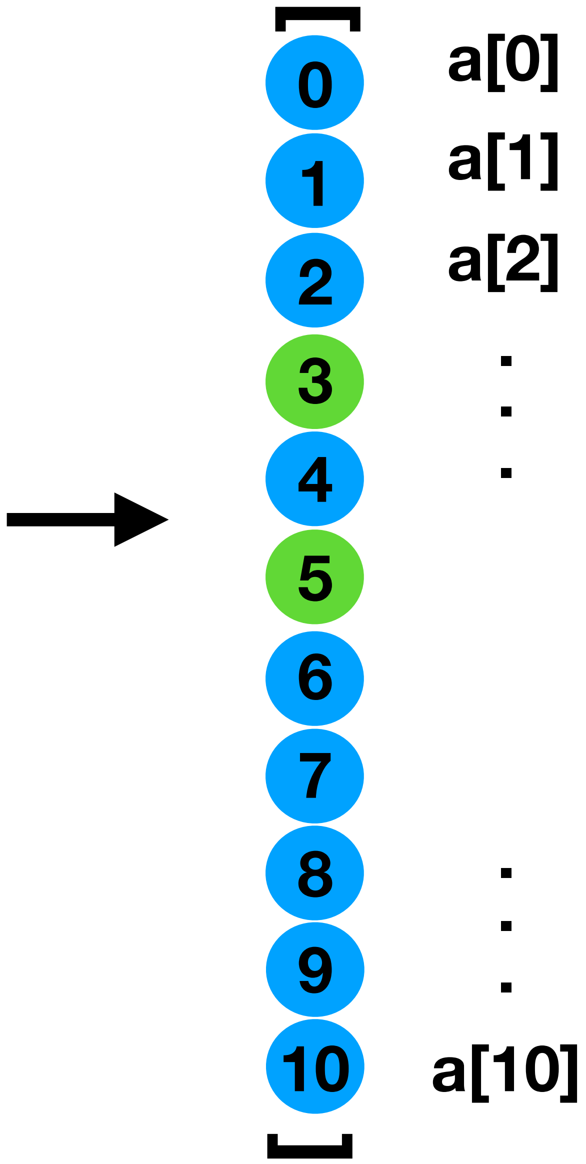
Objects:
(int, float, complex, ...)



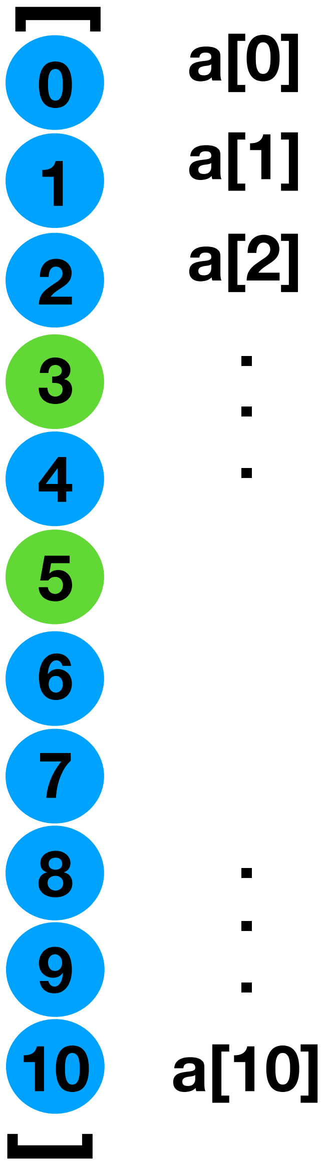
```
a = [
   item_0,
   item_1,
   item_2,
   item_3,
   item_4,
   item_5,
   item_6,
   item_7,
   item_8,
   item_9,
   item_10]
```

Python Sequences

```
a = [
   item_0,
   item_1,
   item_2,
   item_3,
   item_4,
   item_5,
   item_6,
   item_7,
   item_8,
   item_9,
   item_10]
```



Python Loops



A loop allows to a code to be executed repeatedly over a condition. There is two main types of loops:

- For loops:
  The loop is controlled through an iterator as we presented in the last slide. The value of each parameter is assigned to a variable.

- While loops:
  The loop is controlled through an escape condition represented by a boolean (True or False) statement. While something is True I continue, otherwise I stop.

While loops can always replace for loops but can lead to infinite loops if the escape condition is not met.

Python If statements

If statements helps to distinguish cases:

- **if** statement: If a condition is met perform this code
- **elif** statement: Otherwise if this statement is met perform this
- **else** statement: If none of the above applies perform this

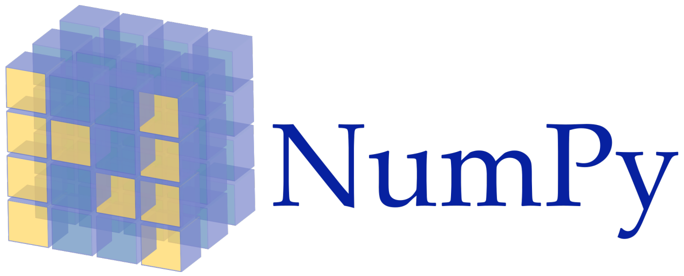
The if statement takes any of the boolean expression and operations and the result should always be either True or False.

Python Modules

```
import matplotlib as mp
```

```
import scipy
```

```
import sklearn
```



```
import numpy as np
```

Summary

09:00 - 10:00 Installing Python and Bornagain

10:00 - 10:05 What is Python

10:05 - 10:40 Python basics

10:40 - 11:00 Coffee break

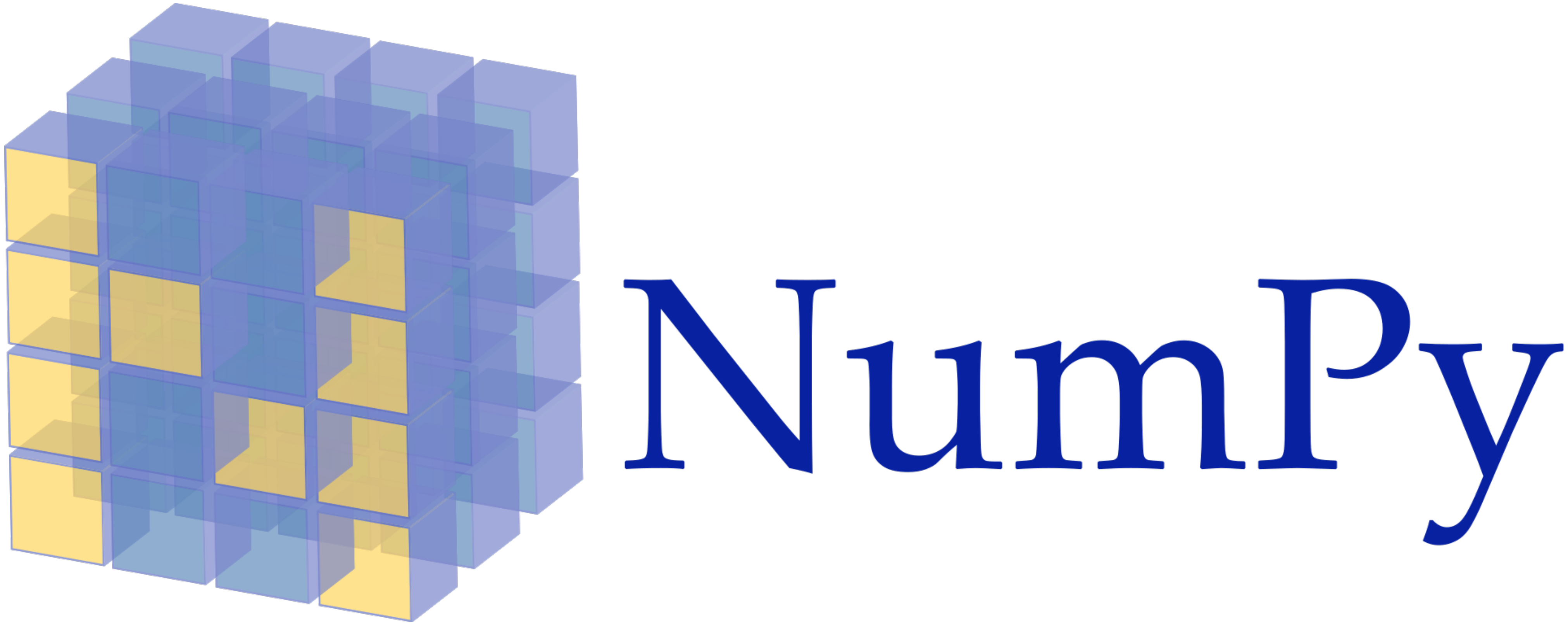
11:00 - 11:30 NumPy introduction

11:30 - 12:00 NumPy advanced

12:00 - 12:30 Matplotlib

12:30 - 13:00 Advanced python

NumPy Introduction



‘ NumPy is the fundamental package for scientific computing with Python
It contains among other things a powerful N-dimensional array object.’

Summary

09:00 - 10:00 Installing Python and Bornagain

10:00 - 10:05 What is Python

10:05 - 10:40 Python basics

10:40 - 11:00 Coffee break

11:00 - 11:30 NumPy introduction

11:30 - 12:00 NumPy advanced

12:00 - 12:30 Matplotlib

12:30 - 13:00 Advanced python

Summary

09:00 - 10:00 Installing Python and Bornagain

10:00 - 10:05 What is Python

10:05 - 10:40 Python basics

10:40 - 11:00 Coffee break

11:00 - 11:30 NumPy introduction

11:30 - 12:00 NumPy advanced

12:00 - 12:30 Matplotlib

12:30 - 13:00 Advanced python

Summary

09:00 - 10:00 Installing Python and Bornagain

10:00 - 10:05 What is Python

10:05 - 10:40 Python basics

10:40 - 11:00 Coffee break

11:00 - 11:30 NumPy introduction

11:30 - 12:00 NumPy advanced

12:00 - 12:30 Matplotlib

12:30 - 13:00 Advanced python