

CHAPITRE 1

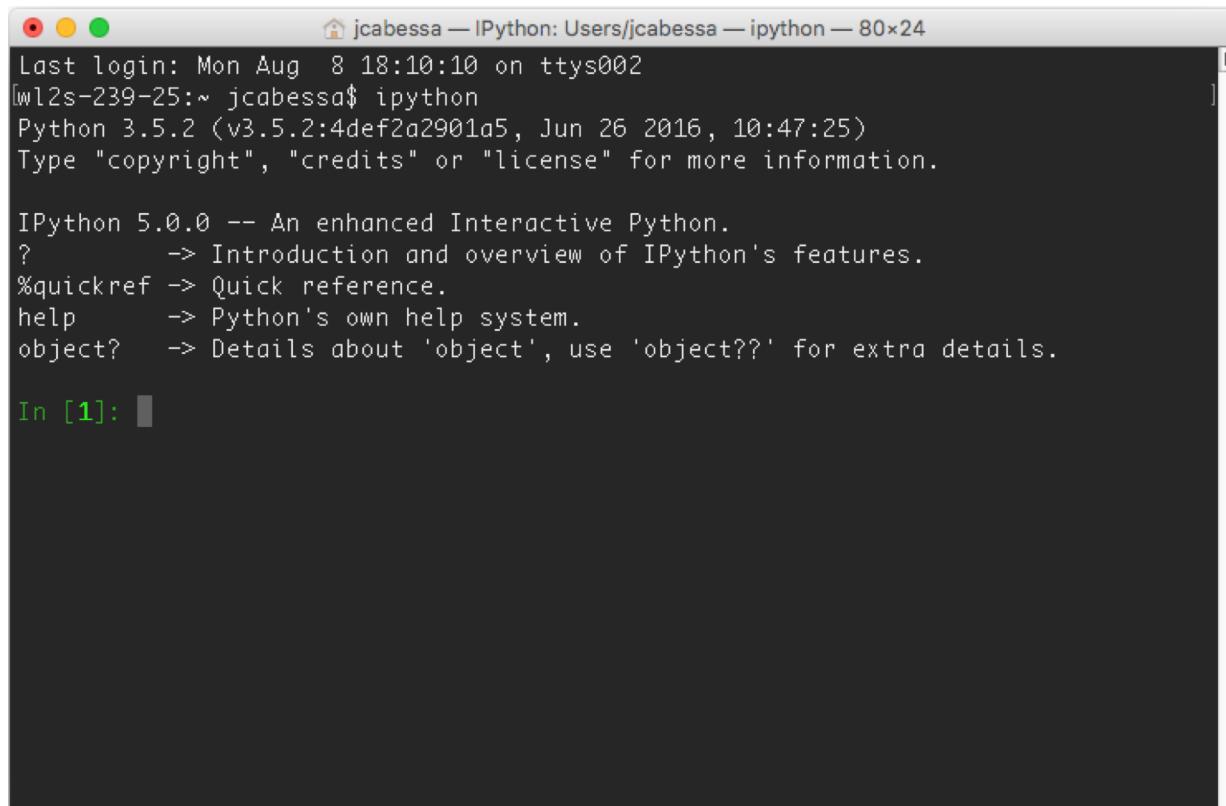
PREMIERS PAS EN PYTHON

Programmation en Python

Lancement de Python

2

- Pour lancer Python, ouvrir un terminal (invite de commandes) et taper « python » ou « ipython ».



A screenshot of a terminal window titled "jcabessa — IPython: Users/jcabessa — ipython — 80x24". The window shows the IPython startup sequence:

```
Last login: Mon Aug  8 18:10:10 on ttys002
[wl2s-239-25:~ jcabessa$ ipython
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 26 2016, 10:47:25)
Type "copyright", "credits" or "license" for more information.

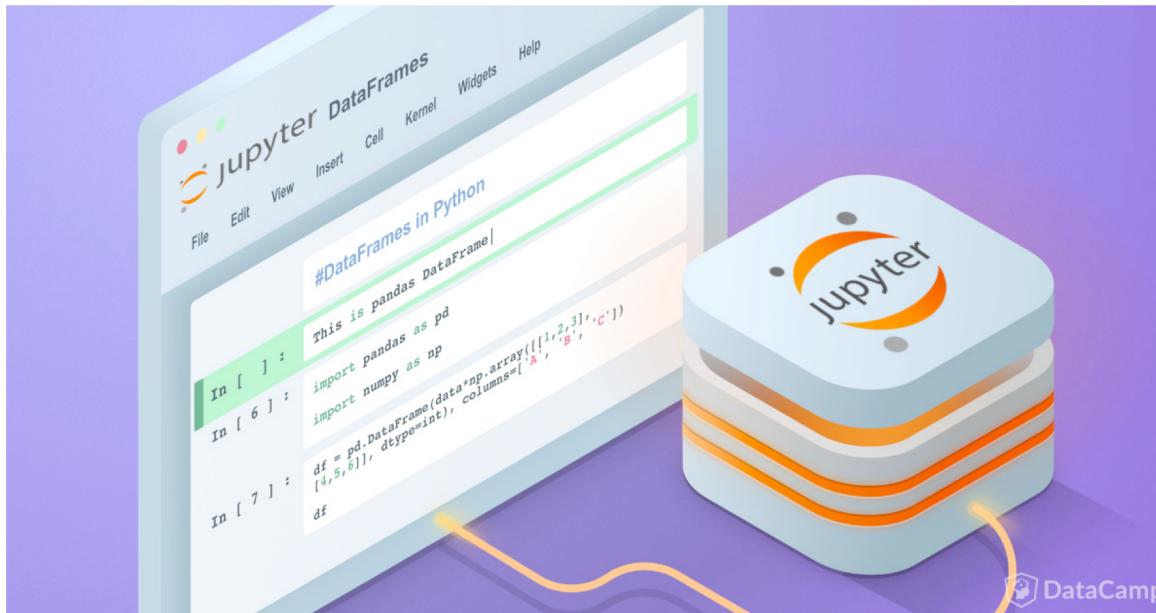
IPython 5.0.0 -- An enhanced Interactive Python.
?          -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help       -> Python's own help system.
object?    -> Details about 'object', use 'object??' for extra details.

In [1]:
```

IDE & Web-based interactive environment

3

- Nous allons utiliser l'environnement de développement (IDE) « Spyder ».
- Il y a également l'application web de développement interactif « Jupyter ».



Entiers et flottants

4

```
>>> # ceci est un commentaire, il est ignoré par Python
>>>
>>> 2+2
4
>>> 2*3
6
>>> 2.5*3.2
8.0
>>> 7/2 # division entière en Python 2 (par en Python 3)
3
>>> 7.0/2 # division réelle
3.5
```

Entiers et flottants

5

```
>>> 10//3 # division entière en Python 3
3
>>> 27//4
6
>>> 2**3 # opérateur de puissance
8
>>> 2**10
1024
>>> 10%3 # opérateur de modulo
1
>>> 27%4
3
```

La fonction « print »

6

```
# En Python 2, on utilise print x
# En Python 3, on utilise print(x) (comme une fonction)
>>> print("Hello world")
Hello world
>>> print("Bonjour")
Bonjour
>>> print(2 + 2)
4
>>> print(3.0 * 5)
15.0
>>>
```

Affectation de variables

7

```
>>> x = 3
>>> y = 4.0
>>> z = "bonjour"
>>>
>>> x
3
>>> y
4.0
>>> z
bonjour
>>>
```

Affectation de variables

8

```
>>> x + y  
7.0  
>>> x * y  
12.0  
>>> z + z  
bonjourbonjour  
>>> x + z  
  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
TypeError: unsupported operand type(s) for +: 'int' and  
'str'  
>>>
```

Expressions booléennes

9

```
>>> 2 < 3  
True  
>>> 7 > 12  
False  
>>> 3 == 4  
False  
>>> 12 == 12  
True  
>>> 3 <= 5  
True  
>>> 3 >= 5  
False
```

Types de données

10

```
>>> type(x)
<class 'int'>

>>> type(y)
<class 'float'>

>>> type(z)
<class 'str'>

>>> type(2 < 3)
<class 'bool'>

>>> type(3 == 4)
<class 'bool'>

>>>
```