

*Collection*  
*Ressources Informatiques*

# PYTHON

**Les fondamentaux du langage  
La programmation pour les scientifiques**

**Matthieu BRUCHER**



**INFORMATIQUE TECHNIQUE**

**eni**  
éditions

2-005-671-2



2-005-671-2




# PYTHON

**Les fondamentaux du langage**  
**La programmation pour les scientifiques**

Certains éléments de cet ouvrage sont téléchargeables à partir du site des Éditions ENI :

<http://www.editions-eni.fr/livres>

Cliquez sur le signe  du cadre **Nos collections**. Cliquez sur la collection **Ressources Informatiques** puis sur **Python - Les fondamentaux du langage - La programmation pour les scientifiques**. Dans l'encadré **En savoir plus**, cliquez sur **Des fichiers complémentaires**.

## Introduction et origines

Chapitre 1

A. Qu'est Python ? . . . . .	7
B. Pourquoi du calcul scientifique sous Python ? . . . . .	13
C. Installation et utilisation . . . . .	14
D. Évolutions futures de Python . . . . .	15

## Le langage Python

Chapitre 2

A. Types prédéfinis . . . . .	19
B. Contrôle de flux. . . . .	38
C. Fonctions . . . . .	42
D. Classes et modules . . . . .	55
E. Gestion des exceptions. . . . .	69
F. Aller plus loin . . . . .	71



## **Les modules indispensables de la bibliothèque standard**

Chapitre 3

A. Le module sys . . . . .	77
B. Le module os. . . . .	78
C. Les modules mathématiques . . . . .	82
D. Mesure de qualité du code . . . . .	86
E. D'autres modules indispensables . . . . .	98

## **D'autres modules de la bibliothèque standard**

Chapitre 4

A. Parallélisme . . . . .	123
B. Expressions régulières . . . . .	137
C. XML . . . . .	142
D. Travailler avec un fichier sur le réseau . . . . .	155
E. Compression . . . . .	157

## **Numpy : une bibliothèque mathématique pour Python**

Chapitre 5

A. Les conteneurs standards . . . . .	169
B. Les fonctions usuelles . . . . .	181
C. Les modules additionnels. . . . .	183

## **Scipy : la boîte à outils scientifiques**

Chapitre 6

A. FFT. . . . .	191
B. Algèbre linéaire . . . . .	193
C. Interpolation . . . . .	198
D. Traitement du signal . . . . .	201
E. Traitement des signaux multidimensionnels . . . . .	210
F. Optimisation . . . . .	219
G. Intégration . . . . .	226
H. Statistiques . . . . .	229

I. Autres modules . . . . . 240

**Graphiques 2D avec Matplotlib** Chapitre 7

A. Création de graphiques . . . . . 249

B. Graphiques avancés . . . . . 257

C. Modification des paramètres par défaut. . . . . 274

**Interface C/C++ avec Python** Chapitre 8

A. CTypeS . . . . . 280

B. Compilation en C à partir d'un script Python . . . . . 290

C. L'interface C Python. . . . . 313

D. Utilisation de wrappers . . . . . 328

E. Une comparaison rapide de vitesse . . . . . 344

Index. . . . . 347

# ▶ Chapitre 1 : Introduction et origines

<b>A. Qu'est Python ?</b>	<b>7</b>
1. Un langage de script	7
2. Un langage orienté objet	8
a. Modularité, souplesse et intuition	8
b. L'orienté objet dans Python	9
3. Un typage dynamique fort	10
4. Une gestion de la mémoire automatique	11
5. Souplesse de style de programmation	12
<b>B. Pourquoi du calcul scientifique sous Python ?</b>	<b>13</b>
<b>C. Installation et utilisation</b>	<b>14</b>
1. Installation	14
2. Utilisation et outils	14
<b>D. Évolutions futures de Python</b>	<b>15</b>