

Prof. Dr.-Ing. Serigne Bira Gueye

Introduction à la programmation en Python

Algorithmique et applications

Presses universitaires de Dakar

Introduction à la programmation en Python

Algorithmique et applications



**Tous droits de production, de reproduction et d'adaptation
réservés pour tous pays**

Dépôt légal : premier trimestre 2026

ISBN : 978-2-494601-75-8

EAN : 9782494601758

Dédicace

À mes parents, Professeur Samba Ndoucoumane GUEYE et Awa FALL,

À mon épouse, Ndiémé NDIAYE,

À mes enfants, Mohammad et Mariam,

À tous mes Professeurs,

À tous mes proches, amis, collègues et étudiants.

Avant-propos

La programmation est devenue, au fil des décennies, une compétence essentielle non seulement pour les informaticiens, mais également pour les chercheurs, les ingénieurs et les professionnels de multiples disciplines.

Ce livre, *Introduction à la programmation en Python*, est le fruit d'une réflexion pédagogique nourrie par plusieurs années d'enseignement et de recherche. Il s'adresse aux élèves, aux étudiants, aux enseignants et à toute personne souhaitant acquérir des bases solides en algorithmique et en programmation, avec une progression allant des notions fondamentales jusqu'aux applications avancées en intelligence artificielle.

L'ouvrage se veut à la fois rigoureux et accessible : rigoureux dans la présentation des concepts, accessible dans l'approche progressive des exemples et des exercices.

Il est organisé en plusieurs parties comprenant des chapitres dont les thèmes sont les suivants :

Partie I : Premiers Pas en Programmation Python

 Chapitre 1 — Premiers programmes : introduction aux bases de la programmation et à l'environnement Python.

 Chapitre 2 — Types élémentaires de données : présentation des types élémentaires (entiers, flottants, booléens, chaînes de caractères)

 Chapitre 3 — Condition et choix : instructions conditionnelles `if/else`, `match/case`.

 Chapitre 4 — Les boucles en Python : boucles `for`, `while`, `break`, `continue`, et structure `else`.

 Chapitre 5 — Fonctions et procédures : sous-programmes, fonctions, procédures, `lambda`.

Partie II : Structures de Données en Python

 Chapitre 6 — Tableaux et listes : tableaux, listes, indexation, slicing, compréhension de listes.

 Chapitre 7 — Dictionnaires, tuples et ensembles : structures de données intégrées et usages typiques.

 Chapitre 8 — Chaînes de caractères : traitement, fonctions utiles, et méthodes associées.

 Chapitre 9 — Fichiers, file handling : lecture/écriture de fichiers texte, CSV, binaires.

Partie III : Programmation Orientée Objet en Python

 Chapitre 10 — Classes et objets : classes, objets, constructeurs, encapsulation.

 Chapitre 11 — Encapsulation et interfaces : Encapsulation, spécificateurs d'accès, propriétés, interfaces: Getters, Setters.

 Chapitre 12 — Héritage et polymorphisme : classes dérivées, surcharge, méthodes spéciales.

 Chapitre 13 — Modules et packages : importations, création de modules, organisation de projets.

 Chapitre 14 — Exceptions et gestion d'erreurs : try/except, finally, levée d'exceptions personnalisées.

 Chapitre 15 — Tests et débogage : assertions, unit tests, debugger, tracebacks.

Partie IV : Calcul Scientifique et Traitement de Données

 , Chapitre 16 — Calcul scientifique et analyse de données avec NumPy : tableaux, opérations vectorielles et matricielles.

 , Chapitre 17 — Analyse et traitement de données tabulaires avec Pandas : séries, dataFrames, filtrage, agrégations, nettoyage.

 , Chapitre 18 — Visualisation de données avec Matplotlib et Seaborn : graphiques de base, personnalisation, visualisation statistique.

 , Chapitre 19 — Calculs scientifiques avancés avec SciPy : optimisation, intégration, équations différentielles.

 , Chapitre 20 — Calcul symbolique avec SymPy : équations, dérivées, intégrales, affichage LaTeX.

 , Chapitre 21 — Traitement d'images avec OpenCV : filtrage, détection de contours, manipulation d'images.

 , Chapitre 22 — Manipulation d'images avec Pillow (PIL) : filtrage, détection de contours, manipulation d'images.

Partie V : Intelligence Artificielle et Apprentissage Automatique

 , Chapitre 23 — Introduction au Machine Learning avec Scikit-Learn : classification, régression, prétraitement.

 , Chapitre 24 — Introduction au Deep Learning avec TensorFlow et Keras : réseaux de neurones, CNN, sauvegarde de modèles.

 , Chapitre 25 — Transfer Learning : réutilisation de modèles pré-entraînés, fine-tuning.

 , Chapitre 26 — Introduction au traitement du langage naturel (NLP) : prétraitement, vectorisation, analyse de sentiments.

 , Chapitre 27 — RNN, LSTM et GRU pour le traitement du langage naturel : séquences, prédiction, traduction simple.

 , Chapitre 28 — Transformers et BERT pour le NLP : attention, modèles HuggingFace, classification, résumé automatique.

Partie VI : Bases de Données et Interfaces Graphiques

 , Chapitre 29 — Fichiers avancés et formats de données : JSON, XML, sérialisation avec pickle.

 , Chapitre 30 — Bases de données relationnelles avec SQLite : création de tables, insertion, requêtes, mises à jour.

 , Chapitre 31 — Connexion à MySQL et PostgreSQL : connecteurs, exécution de requêtes paramétrées, gestion des données.

 , Chapitre 32 — Utilisation de SQLAlchemy : ORM, modèles, opérations CRUD, compatibilité multi-SGBD.

 , Chapitre 33 — Interfaces graphiques avec Tkinter : widgets de base, événements, interaction avec une base de données.

 , Chapitre 34 — Interfaces graphiques avancées avec PyQt5 : Qt Designer, signaux et slots, formulaires connectés.

 , Chapitre 35 — Applications multiplateformes avec Kivy : layouts, événements tactiles, lier Kivy à SQLite.

 , Chapitre 36 — Consommation d'API Web en Python : utilisation de requêtes, lecture et écriture de fichiers JSON.

Partie VII : Conclusion et Perspectives

 , Chapitre 37 — Conclusion et Perspectives : résumé des acquis, avenir de Python et de l'IA, message final de l'auteur.

Remerciements

Je remercie chaleureusement tous ceux qui ont contribué à la réalisation de ce livre, en particulier mes collègues Dr. Jean Paul Latyr Faye et Prof. Kharouna Talla.

Je souhaite également exprimer ma reconnaissance à ma femme, Ndiémé Ndiaye, et à mes enfants, Mouhammad et Mariam, pour leur soutien constant et leurs encouragements.

Table des matières

Avant-propos	i
Remerciements	iv
Table des matières	v
 Première partie Premiers pas en programmation Python	 19
1 Premiers programmes	21
1.1 Objectifs du chapitre	21
1.2 Qu'est-ce qu'un algorithme ?	21
1.3 Installer Python et préparer l'environnement	21
1.4 Premier programme	22
1.5 Éléments de base d'un programme	22
1.5.1 Constantes	22
1.5.2 Commentaires	22
1.5.3 Variables et affectations	23
1.5.4 Variables et types de données	24
1.5.5 Identifiants	24
1.6 Entrée et sortie des données (input, output)	24
1.6.1 Entrée des données (input)	24
1.6.2 Lecture de plusieurs valeurs	25
1.6.3 Sortie des données (output)	25
1.6.4 Lecture au clavier et affichage formaté	25
1.7 Application: calcul du périmètre et de l'aire d'un rectangle	26
1.8 Mini-projet	26
1.9 Exercices	27
Bibliographie	27
 2 Types de données élémentaires	 29
2.1 Objectifs du chapitre	29
2.2 Les différents types de base	29

Table des matières

2.3	Le type entier (int)	29
2.3.1	Littéraux et bases	30
2.3.2	Opérateurs d'entiers	30
2.4	Le type booléen (bool)	31
2.4.1	Opérateurs logiques	31
2.4.2	Valeurs logiques et vérité des objets	31
2.4.3	Opérateurs relationnels	32
2.5	Flottants : float	32
2.5.1	Arrondi et formatage	32
2.6	Chaînes : str	33
2.6.1	Unicode, immutabilité et littéraux	33
2.6.2	Échappements et chaînes brutes	33
2.6.3	Indexation, tranches et longueur	33
2.6.4	Méthodes utiles et f-strings	34
2.6.5	Bonnes pratiques de manipulation de strings	34
2.7	La valeur None	34
2.8	Accès au type d'une donnée	35
2.9	Conversions de type	35
2.9.1	Erreurs fréquentes de conversion	35
2.10	Résumé des opérateurs arithmétiques	35
2.11	Applications guidées	36
2.12	Exercices	36
	Bibliographie	37
3	Condition et choix	39
3.1	Objectifs du chapitre	39
3.2	Structures de branchement	39
3.3	La condition if: branchement conditionnel	39
3.3.1	Instruction if simple	40
3.3.2	Instruction if/else	41
3.3.3	Instruction if/elif/else (choix multiples)	41
3.3.4	Conditions imbriquées	41
3.3.5	Expression conditionnelle (opérateur ternaire)	42
3.3.6	Bonnes pratiques	42
3.3.7	Conditions en Pascal et C++	42
3.4	La structure match/case (Choix, Selon, Cas où)	42
3.4.1	Syntaxe en Python	42
3.4.2	Exemple : choix du mois	43
3.5	Applications guidées	43
3.6	Le saut inconditionnel (goto)	44
3.7	Conclusion	44

Table des matières

3.8	Exercices	44
	Bibliographie	45
4	Les boucles en Python	47
4.1	Objectifs du chapitre	47
4.2	Définition	47
4.3	La boucle for	47
4.3.1	La boucle for avec range()	47
4.3.2	Variantes de range()	48
4.3.3	Boucle for sur séquence	48
4.3.4	Boucles imbriquées	48
4.4	La boucle while	49
4.5	Instructions spéciales	49
4.6	Bonnes pratiques	50
4.7	Applications guidées	50
4.8	Résumé	51
4.9	Exercices	51
	Bibliographie	52
5	Fonctions et procédures	53
5.1	Introduction	53
5.2	Fonction et procédure en Pascal	53
5.3	Fonction en C++	54
5.4	Fonction en Python	54
5.4.1	Définition d'une fonction	54
5.4.2	Portée (scope) des variables	54
5.4.3	Paramètres nommés et valeur par défaut	55
5.4.4	Retourner plusieurs valeurs	55
5.5	Fonctions lambda (anonymes)	56
5.6	Fonctions comme paramètres	56
5.7	Applications guidées	56
5.8	Exercices	57
	Bibliographie	57
Deuxième partie Structures de données en Python		59
6	Tableaux et listes	61
6.1	Introduction	61
6.2	Comparaison Pascal/C++ et Python	61
6.3	Création et initialisation de listes	61
6.4	Indexation et parcours	62

Table des matières

6.5	Slicing: tranches, découpage de listes	62
6.6	Modification et méthodes utiles	63
6.7	Listes multidimensionnelles	63
6.8	Recherche et tri	64
6.9	Introduction à NumPy pour les tableaux scientifiques	64
6.10	Applications guidées	65
6.11	Comparaison avec les tableaux en Pascal et C++	65
6.12	Exercices	65
	Bibliographie	66
7	Dictionnaires, tuples et ensembles	67
7.1	Introduction	67
7.2	Dictionnaires	67
7.2.1	Dictionnaire (dict) en Python	68
7.2.2	Opérations courantes sur les dictionnaires	69
7.2.3	Structures imbriquées	69
7.3	Tuples	69
7.3.1	Syntaxe générale	70
7.3.2	Création de tuples	70
7.3.3	Accès aux éléments	70
7.3.4	Immutabilité	70
7.3.5	Désassemblage (unpacking)	71
7.3.6	Utilité des tuples	71
7.4	Ensembles	71
7.4.1	Pascal	71
7.4.2	Python: set	72
7.4.3	Syntaxe des ensembles en Python	72
7.4.4	Opérations ensemblistes en Python	72
7.5	Applications guidées	73
7.6	Exercices	74
	Bibliographie	74
8	Chaînes de caractères	77
8.1	Introduction	77
8.2	Création et affectation de strings	77
8.3	Concaténation et répétition	77
8.4	Indexation et slicing de strings	78
8.5	Méthodes utiles	78
8.6	Boucle sur caractères	78
8.7	Chaînes multilignes et formatage	79
8.8	Applications guidées	79

Table des matières

8.9	Exercices	80
	Bibliographie	80
9	Fichier, file handling	83
9.1	Introduction	83
9.2	Ouverture et fermeture d'un fichier	83
9.2.1	Modes d'ouverture	84
9.2.2	Utilisation du <code>with</code>	84
9.3	Lecture et écriture d'un fichier	85
9.4	Gestion des exceptions	85
9.5	Fichiers binaires	85
9.6	Manipulation de fichiers CSV	86
9.6.1	Lecture/écriture manuelle	86
9.6.2	Lecture/écriture avec le module <code>csv</code>	86
9.7	Applications guidées	87
9.8	Exercices	88
	Bibliographie	88
Troisième partie Programmation orientée objet en Python		91
10	Classe et objets	93
10.1	Introduction	93
10.2	Classe et objets en Python	93
10.2.1	Syntaxe générale d'une classe en Python	93
10.2.2	Création d'une classe et d'objets	94
10.3	Comparaisons avec d'autres langages	95
10.4	Application guidée	97
10.5	Exercices	97
	Bibliographie	98
11	Encapsulation et interfaces	99
11.1	Introduction	99
11.2	Encapsulation en Python	99
11.3	Comparaison avec Pascal et C++	100
11.4	Accès indirect, interfaces: <code>Getters</code> et <code>Setters</code>	100
11.5	Approche pythonique : décorateur <code>@property</code>	101
11.6	Exemple complet : contrôle d'accès	102
11.7	Applications guidées	102
11.8	Exercices	103
	Bibliographie	103

Table des matières

12 Héritage et polymorphisme	105
12.1 Introduction	105
12.2 Héritage	105
12.2.1 Héritage simple	105
12.2.2 Redéfinition et utilisation de <code>super()</code>	105
12.2.3 Héritage multiple	106
12.3 Polymorphisme	106
12.3.1 Polymorphisme simple	106
12.3.2 Polymorphisme avancé	107
12.4 Classes et méthodes abstraites	107
12.5 Méthodes de classe et méthodes statiques	107
12.6 Exemple complet : formes géométriques	108
12.7 Applications guidées	109
12.8 Exercices	109
Bibliographie	109
13 Modules et packages	111
13.1 Introduction	111
13.2 Comparaisons avec d'autres langages	111
13.3 Création d'un module en Python	112
13.4 Importer un module	112
13.4.1 Import sélectif	112
13.4.2 Alias	112
13.5 Variable spéciale <code>__name__</code>	113
13.6 Organisation en packages	113
13.7 Import relatif et absolu	113
13.8 Organisation avancée de projets	113
13.9 Import dynamique avec <code>importlib</code>	114
13.10 Utilisation de modules externes	114
13.11 Modules standards utiles	114
13.12 Création et distribution d'un package	114
13.13 Applications guidées	115
13.14 Exercices	116
Bibliographie	116
14 Exceptions et gestion d'erreurs	119
14.1 Introduction	119
14.2 Erreurs vs exceptions	119
14.3 Bloc <code>try/except</code>	119
14.4 Bloc <code>else</code> et <code>finally</code>	119
14.5 Lever une exception	120

Table des matières

14.6	Exceptions personnalisées	120
14.7	Comparaison	120
14.8	Applications guidées	121
14.9	Exceptions courantes	122
14.10	Bonnes pratiques	122
14.11	Exercices	122
	Bibliographie	123
15	Tests et débogage	125
15.1	Introduction	125
15.2	Test vs débogage	125
15.3	Tests manuels	125
15.4	Utilisation de print() pour déboguer	125
15.5	Utilisation de pdb	125
15.6	Assert	126
15.7	Tests unitaires avec unittest	126
15.8	Comparaison avec Pascal et C++	127
15.9	Applications guidées	127
15.10	Exercices	127
	Bibliographie	127
Quatrième partie Calcul scientifique et traitement de données		129
16	Calcul scientifique et analyse de données avec NumPy	131
16.1	Introduction	131
16.2	Installation	131
16.3	Création de tableaux NumPy	131
16.4	Tableaux prédéfinis et spéciaux	131
16.5	Accès et modification des éléments	132
16.6	Opérations vectorielles et élémentaires	132
16.7	Fonctions universelles	132
16.8	Opérations sur les matrices	133
16.9	Statistiques et analyse de données	133
16.10	Indexation booléenne	133
16.11	Résolution de systèmes d'équations linéaires	134
16.12	Applications guidées	134
16.13	Exercices	134
	Bibliographie	135
17	Analyse et traitement de données tabulaires avec Pandas	137
17.1	Introduction	137

Table des matières

17.2	Installation	137
17.3	Création d'une série et d'un <code>dataFrame</code>	137
17.4	Lecture et écriture de fichiers	138
17.4.1	CSV	138
17.4.2	Excel	138
17.5	Exploration des données	138
17.6	Sélection et filtrage	138
17.7	Modification des données	139
17.8	Gestion des valeurs manquantes	139
17.9	Opérations statistiques	139
17.10	Groupement et agrégation	140
17.11	Fusion et jointure de <code>dataFrames</code>	140
17.12	Applications guidées	140
17.13	Exercices	140
	Bibliographie	141
18	Visualisation de données avec Matplotlib et Seaborn	143
18.1	Introduction	143
18.2	Installation	143
18.3	Matplotlib — graphiques de base	143
18.3.1	Courbe simple	143
18.3.2	Histogramme	143
18.3.3	Diagramme en barres	144
18.4	Seaborn — graphiques avancés	144
18.4.1	Exemple avec <code>dataframe</code> Pandas	144
18.4.2	Nuage de points	144
18.4.3	Distribution des données	145
18.5	Personnalisation des graphiques	145
18.6	Applications guidées	145
18.7	Exercices	146
	Bibliographie	146
19	Calculs scientifiques avancés avec SciPy	147
19.1	Introduction	147
19.2	Installation	147
19.3	Optimisation	147
19.4	Intégration numérique	147
19.5	Résolution d'équations différentielles	148
19.6	Traitement du signal	148
19.7	Statistiques avancées	148
19.8	Applications guidées	149

Table des matières

19.9 Exercices	149
Bibliographie	149
20 Calcul symbolique avec SymPy	151
20.1 Introduction	151
20.2 Installation	151
20.3 Déclaration de symboles	151
20.4 Manipulation d'expressions	151
20.5 Résolution d'équations	152
20.6 Dérivées et intégrales	152
20.7 Équations différentielles	152
20.8 Valeurs numériques	152
20.9 Affichage en LaTeX	153
20.10 Applications guidées	153
20.11 Exercices	153
Bibliographie	153
21 Traitement d'images avec OpenCV	155
21.1 Introduction	155
21.2 Installation	155
21.3 Lecture et affichage d'une image	155
21.4 Conversion en niveaux de gris	155
21.5 Redimensionnement et recadrage	156
21.6 Dessin sur une image	156
21.7 Détection de contours	156
21.8 Flou et filtrage	157
21.9 Lecture de vidéo	157
21.10 Applications guidées	157
21.11 Exercices	158
Bibliographie	158
22 Manipulation d'images avec Pillow (PIL)	159
22.1 Introduction	159
22.2 Installation	159
22.3 Chargement et affichage d'une image	159
22.4 Informations sur l'image	159
22.5 Conversion de format	160
22.6 Redimensionnement et rotation	160
22.7 Recadrage	160
22.8 Conversion en niveaux de gris	160
22.9 Ajout de texte sur une image	160

Table des matières

22.10	Application de filtres	161
22.11	Applications guidées	161
22.12	Exercices	161
	Bibliographie	161
 Cinquième partie Intelligence artificielle et apprentissage automatique		163
23	Introduction au Machine Learning avec Scikit-Learn	165
23.1	Introduction	165
23.2	Installation	165
23.3	Étapes d'un projet de Machine Learning	165
23.4	Prétraitement des données	166
23.5	Entraînement avec un autre modèle	166
23.6	Applications guidées	167
23.7	Exercices	167
	Bibliographie	167
24	Introduction au Deep Learning avec TensorFlow et Keras	169
24.1	Introduction	169
24.2	Installation	169
24.3	Structure d'un réseau de neurones	169
24.4	Exemple : classification de chiffres manuscrits (MNIST)	170
24.5	Réseaux convolutifs (CNN) pour la vision	171
24.6	Sauvegarde et chargement de modèles	171
24.7	Applications guidées	171
24.8	Exercices	172
	Bibliographie	172
25	Transfer Learning	173
25.1	Introduction	173
25.2	Principe	173
25.3	Exemple avec MobileNetV2	173
25.4	Entraînement sur de nouvelles données	174
25.5	Affinage (Fine-Tuning)	175
25.6	Applications guidées	175
25.7	Exercices	175
	Bibliographie	176
26	Introduction au traitement du langage naturel (NLP)	177
26.1	Introduction	177

Table des matières

26.2	Bibliothèques utiles	177
26.3	Prétraitement du texte	177
26.4	Vectorisation du texte	178
26.5	Exemple : analyse de sentiments avec Keras	178
26.6	Applications guidées	179
26.7	Exercices	179
	Bibliographie	180
27	RNN, LSTM et GRU pour le traitement du langage naturel	181
27.1	Introduction	181
27.2	Exemple simple avec un RNN	181
27.3	Exemple avec LSTM	182
27.4	Exemple avec GRU	182
27.5	Applications guidées	183
27.6	Exercices	183
	Bibliographie	183
28	Transformers et BERT pour le NLP	185
28.1	Introduction	185
28.2	Principe de l'attention	185
28.3	Installation de la bibliothèque Hugging Face	185
28.4	Exemple : analyse de sentiments avec BERT	186
28.5	Exemple : remplissage de texte (Masked Language Modeling)	186
28.6	Exemple : génération de texte avec GPT-2	186
28.7	Applications guidées	187
28.8	Exercices	187
	Bibliographie	187
Sixième partie Bases de données et applications avancées		189
29	Fichiers avancés et formats de données	191
29.1	Introduction	191
29.2	Fichiers JSON	191
29.3	Fichiers XML	191
29.4	Sérialisation avec pickle	192
29.5	Comparaison des formats	192
29.6	Applications guidées	193
29.7	Exercices	193
	Bibliographie	194
30	Bases de données relationnelles avec SQLite	195

Table des matières

30.1	Introduction	195
30.2	Connexion à une base	195
30.3	Création d'une table	195
30.4	Insertion de données	196
30.5	Lecture de données	196
30.6	Mise à jour et suppression	196
30.7	Fermeture de la connexion	196
30.8	Requêtes paramétrées	197
30.9	Applications guidées	197
30.10	Exercices	197
	Bibliographie	197
31	Connexion à MySQL et PostgreSQL	199
31.1	Introduction	199
31.2	Connexion à MySQL	199
31.2.1	Installation du connecteur	199
31.2.2	Exemple de connexion	199
31.3	Connexion à PostgreSQL	200
31.3.1	Installation du connecteur	200
31.3.2	Exemple de connexion	200
31.4	Exécution de requêtes SQL	200
31.5	Requêtes paramétrées	201
31.6	Différences entre MySQL et PostgreSQL	201
31.7	Applications guidées	201
31.8	Exercices	202
	Bibliographie	202
32	Utilisation de SQLAlchemy	203
32.1	Introduction	203
32.2	Installation	203
32.3	Connexion à une base SQLite	203
32.4	Définition d'un modèle	203
32.5	Création de la table	204
32.6	Insertion de données	204
32.7	Lecture de données	204
32.8	Mise à jour et suppression	204
32.9	Connexion à MySQL ou PostgreSQL	205
32.9.1	MySQL	205
32.9.2	PostgreSQL	205
32.10	Applications guidées	205
32.11	Exercices	206

Table des matières

Bibliographie	206
33 Interfaces graphiques avec Tkinter	207
33.1 Introduction	207
33.2 Création d'une fenêtre simple	207
33.3 Widgets de base	207
33.3.1 Label et button	207
33.3.2 Entry (champ de saisie)	208
33.4 Organisation avec frame	208
33.5 Lier Tkinter à une base de données	208
33.6 Applications guidées	209
33.7 Exercices	209
Bibliographie	210
34 Interfaces graphiques avancées avec PyQt5	211
34.1 Introduction	211
34.2 Installation	211
34.3 Fenêtre simple avec PyQt5	211
34.4 Widgets de base	211
34.5 Champs de saisie	212
34.6 Organisation avec layouts	213
34.7 Création d'une interface avec Qt Designer	213
34.8 Liaison PyQt5 et base de données SQLite	213
34.9 Applications guidées	213
34.10 Exercices	214
Bibliographie	214
35 Création d'applications multiplateformes avec Kivy	217
35.1 Introduction	217
35.2 Installation	217
35.3 Première application Kivy	217
35.4 Boutons et événements	217
35.5 Organisation avec BoxLayout	218
35.6 Utilisation d'un fichier KV	218
35.6.1 fichier main.py	218
35.6.2 fichier monapp.kv	219
35.7 Connexion à une base de données SQLite	219
35.8 Applications guidées	220
35.9 Exercices	220
Bibliographie	220
36 Consommation d'API Web en Python	221

Table des matières

36.1	Introduction	221
36.2	Installation de Requests	221
36.3	Effectuer une requête GET	221
36.4	Effectuer une requête POST	221
36.5	Utilisation des paramètres de requête	222
36.6	Lecture et écriture de fichiers JSON	222
36.7	Exemple : application météo avec une API	222
36.8	Intégration avec Tkinter ou PyQt5	223
36.9	Applications guidées	223
36.10	Exercices	224
	Bibliographie	224
	 Conclusion	 225
	 Après-propos	 229
	Mot de l'auteur	229