

Gérard Swinnen

La référence
en apprentissage de
la programmation !
3^e édition

Apprendre à programmer avec Python 3

Avec 60 pages d'exercices corrigés !

Objet · Multithreading · Bases de données · Événements
Programmation web · Programmation réseau · Unicode
Impression PDF · Python 2.7 & 3.2 · Tkinter · CherryPy



EYROLLES

Gérard Swinnen

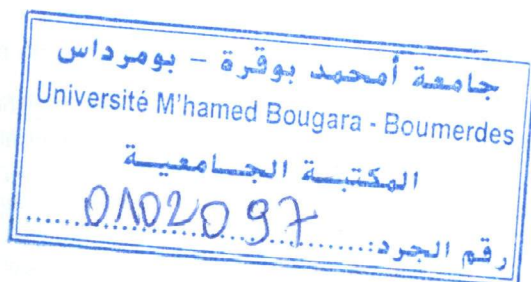


Apprendre à programmer avec Python 3

Avec 60 pages d'exercices corrigés !

**Objet • Multithreading • Bases de données • Événements
Programmation web • Programmation réseau • Unicode
Impression PDF • Python 2.7 & 3.2 • Tkinter • CherryPy**

3^e édition



1exp.

EYROLLES

Table des matières

1. À L'ÉCOLE DES SORCIERS	1
Boîtes noires et pensée magique • 1	
Magie blanche, magie noire • 2	
La démarche du programmeur • 3	
Langage machine, langage de programmation • 4	
Édition du code source – Interprétation • 6	
Mise au point d'un programme – Recherche des erreurs (debug) • 6	
Erreurs de syntaxe • 6	
Erreurs sémantiques • 7	
Erreurs à l'exécution • 7	
Recherche des erreurs et expérimentation • 8	
2. PREMIERS PAS	9
Calculer avec Python • 9	
Données et variables • 11	
Noms de variables et mots réservés • 12	
Affectation (ou assignation) • 12	
Afficher la valeur d'une variable • 13	
Typage des variables • 14	
Affectations multiples • 15	
Opérateurs et expressions • 15	
Priorité des opérations • 16	
Composition • 17	
3. CONTRÔLE DU FLUX D'EXÉCUTION	19
Séquence d'instructions • 19	
Sélection ou exécution conditionnelle • 20	
Opérateurs de comparaison • 21	
Instructions composées – blocs d'instructions • 21	
Instructions imbriquées • 22	
Quelques règles de syntaxe Python • 22	
Les limites des instructions et des blocs sont définies par la mise en page • 23	
Instruction composée : en-tête, double point, bloc d'instructions indenté • 23	
Les espaces et les commentaires sont normalement ignorés • 24	
4. INSTRUCTIONS RÉPÉTITIVES	25
Réaffectation • 25	
Répétitions en boucle – L'instruction while • 26	
Commentaires • 26	
Remarques • 27	
Élaboration de tables • 27	
Construction d'une suite mathématique • 28	
Premiers scripts, ou comment conserver nos programmes • 29	
Problèmes éventuels liés aux caractères accentués • 32	
5. PRINCIPAUX TYPES DE DONNÉES	35
Les données numériques • 35	
Le type integer • 35	
Le type float • 37	
Les données alphanumériques • 38	
Le type string • 39	
Remarques • 40	
Triple quotes • 40	
Accès aux caractères individuels d'une chaîne • 40	
Opérations élémentaires sur les chaînes • 41	
Les listes (première approche) • 42	
6. FONCTIONS PRÉDÉFINIES	47
La fonction print() • 47	
Interaction avec l'utilisateur : la fonction input() • 47	
Importer un module de fonctions • 48	
Un peu de détente avec le module turtle • 50	
Véracité/fausseté d'une expression • 51	
Révision • 53	
Contrôle du flux – utilisation d'une liste simple • 53	
Boucle while – instructions imbriquées • 54	
7. FONCTIONS ORIGINALES	57
Définir une fonction • 57	
Fonction simple sans paramètres • 58	
Fonction avec paramètre • 59	
Utilisation d'une variable comme argument • 60	
Remarque importante • 60	
Fonction avec plusieurs paramètres • 61	
Notes • 61	
Variables locales, variables globales • 62	
Vraies fonctions et procédures • 64	
Notes • 65	
Utilisation des fonctions dans un script • 66	
Notes • 66	
Modules de fonctions • 67	
Typage des paramètres • 72	
Valeurs par défaut pour les paramètres • 72	
Arguments avec étiquettes • 73	

8. UTILISATION DE FENÊTRES ET DE GRAPHISMES ... 75**Interfaces graphiques (GUI) • 75****Premiers pas avec tkinter • 75**

Examinons à présent plus en détail chacune des lignes de commandes exécutées • 76

Programmes pilotés par des événements • 79

Exemple graphique : tracé de lignes dans un canevas • 81

Exemple graphique : deux dessins alternés • 84

Exemple graphique : calculatrice minimaliste • 86

Exemple graphique : détection et positionnement d'un clic de souris • 88

Les classes de widgets tkinter • 89**Utilisation de la méthode() pour contrôler la disposition des widgets • 90****Composition d'instructions pour écrire un code plus compact • 94****Modification des propriétés d'un objet – Animation • 96****Animation automatique – Récursivité • 99****9. MANIPULER DES FICHIERS 103****Utilité des fichiers • 103****Travailler avec des fichiers • 104****Noms de fichiers – le répertoire courant • 105****Les deux formes d'importation • 106****Écriture séquentielle dans un fichier • 107**

Notes • 107

Lecture séquentielle d'un fichier • 108

Notes • 108

L'instruction break pour sortir d'une boucle • 109**Fichiers texte • 110**

Remarques • 111

Enregistrement et restitution de variables diverses • 112**Gestion des exceptions : les instructions try – except – else • 113****10. APPROFONDIR LES STRUCTURES DE DONNÉES 117****Le point sur les chaînes de caractères • 117**

Indiçage, extraction, longueur • 117

Extraction de fragments de chaînes • 118

Concaténation, répétition • 119

Parcours d'une séquence : l'instruction for ... in ... • 120

Appartenance d'un élément à une séquence : l'instruction in utilisée seule • 121

Les chaînes sont des séquences non modifiables • 122

Les chaînes sont comparables • 123

La norme Unicode • 123

Séquences d'octets : le type bytes • 125

L'encodage Utf-8 • 127

Conversion (encodage/décodage) des chaînes • 128

Conversion d'une chaîne bytes en chaîne string • 128

Conversion d'une chaîne string en chaîne bytes • 128

Conversions automatiques lors du traitement des fichiers • 129

Cas des scripts Python • 130

Accéder à d'autres caractères que ceux du clavier • 131

Les chaînes sont des objets • 132

Fonctions intégrées • 134

Formatage des chaînes de caractères • 134

Formatage des chaînes « à l'ancienne » • 135

Le point sur les listes • 137

Définition d'une liste – accès à ses éléments • 137

Les listes sont modifiables • 137

Les listes sont des objets • 138

Techniques de slicing avancé pour modifier une liste • 139
Insertion d'un ou plusieurs éléments n'importe où dans une liste • 139

Suppression / remplacement d'éléments • 140

Création d'une liste de nombres à l'aide de la fonction range() • 140

Parcours d'une liste à l'aide de for, range() et len() • 141

Une conséquence importante du typage dynamique • 141

Opérations sur les listes • 142

Test d'appartenance • 142

Copie d'une liste • 142

Petite remarque concernant la syntaxe • 143

Nombres aléatoires – histogrammes • 144

Tirage au hasard de nombres entiers • 146**Les tuples • 147****Opérations sur les tuples • 148****Les dictionnaires • 149**

Création d'un dictionnaire • 149

Opérations sur les dictionnaires • 150

Test d'appartenance • 150

Les dictionnaires sont des objets • 150

Parcours d'un dictionnaire • 151

Les clés ne sont pas nécessairement des chaînes de caractères • 152

Les dictionnaires ne sont pas des séquences • 153

Construction d'un histogramme à l'aide d'un dictionnaire • 154

Contrôle du flux d'exécution à l'aide d'un dictionnaire • 155

11. CLASSES, OBJETS, ATTRIBUTS 159**Utilité des classes • 159****Définition d'une classe élémentaire • 160****Attributs (ou variables) d'instance • 162****Passage d'objets comme arguments dans l'appel d'une fonction • 163****Similitude et unicité • 163****Objets composés d'objets • 164****Objets comme valeurs de retour d'une fonction • 166****Modification des objets • 166**

- 12. CLASSES, MÉTHODES, HÉRITAGE 167**
 Définition d'une méthode • 168
 Définition concrète d'une méthode dans un script • 169
 Essai de la méthode, dans une instance quelconque • 169
 La méthode constructeur • 170
 Exemple • 170
 Espaces de noms des classes et instances • 174
 Héritage • 175
 Héritage et polymorphisme • 176
 Commentaires • 178
 Modules contenant des bibliothèques de classes • 181
- 13. CLASSES ET INTERFACES GRAPHIQUES 185**
 Code des couleurs : un petit projet bien encapsulé • 185
 Cahier des charges de notre programme • 186
 Mise en œuvre concrète • 186
 Commentaires • 187
 Petit train : héritage, échange d'informations entre objets • 189
 Cahier des charges • 190
 Implémentation • 190
 Commentaires • 191
 OscilloGraphe : un widget personnalisé • 192
 Expérimentation • 194
 Cahier des charges • 195
 Implémentation • 195
 Curseurs : un widget composite • 197
 Présentation du widget Scale • 197
 Construction d'un panneau de contrôle à trois curseurs • 198
 Commentaires • 200
 Propagation des événements • 202
 Intégration de widgets composites dans une application synthèse • 202
 Commentaires • 204
- 14. ET POUR QUELQUES WIDGETS DE PLUS 211**
 Les boutons radio • 211
 Commentaires • 212
 Utilisation de cadres pour la composition d'une fenêtre • 213
 Commentaires • 214
 Comment déplacer des dessins à l'aide de la souris • 215
 Commentaires • 217
 Widgets complémentaires, widgets composites • 219
 Combo box simplifié • 219
 Commentaires • 221
 Le widget Text assorti d'un ascenseur • 222
 Gestion du texte affiché • 223
 Commentaires • 224
 Canevas avec barres de défilement • 225
 Commentaires • 228
 Application à fenêtres multiples – paramétrage implicite • 229
 Commentaires • 232
- Barres d'outils – expressions lambda • 233
 Métaprogrammation – expressions lambda • 234
 Passage d'une fonction (ou d'une méthode) comme argument • 235
- Fenêtres avec menus • 236**
 Cahier des charges de l'exercice • 237
 Première ébauche du programme • 237
 Analyse du script • 238
 Ajout de la rubrique Musiciens • 240
 Analyse du script • 241
 Ajout de la rubrique Peintres • 241
 Analyse du script • 242
 Ajout de la rubrique Options • 242
 Menu avec cases à cocher • 243
 Menu avec choix exclusifs • 244
 Contrôle du flux d'exécution à l'aide d'une liste • 245
 Présélection d'une rubrique • 246
- 15. ANALYSE DE PROGRAMMES CONCRETS 247**
 Jeu des bombardes • 247
 Prototypage d'une classe Canon • 247
 Commentaires • 252
 Ajout de méthodes au prototype • 252
 Commentaires • 254
 Développement de l'application • 254
 Commentaires • 260
 Développements complémentaires • 260
 Commentaires • 265
 Jeu de Ping • 265
 Principe • 266
 Programmation • 266
 Cahier des charges du logiciel à développer • 267
- 16. GESTION D'UNE BASE DE DONNÉES 271**
 Les bases de données • 271
 SGBDR – Le modèle client/serveur • 272
 Le langage SQL • 273
 SQLite • 273
 Création de la base de données – Objets « connexion » et « curseur » • 274
 Connexion à une base de données existante • 276
 Recherches sélectives dans une base de données • 278
 La requête select • 279
 Ébauche d'un logiciel client pour PostgreSQL • 281
 Décrire la base de données dans un dictionnaire d'application • 282
 Définir une classe d'objets-interfaces • 284
 Commentaires • 286
 Construire un générateur de formulaires • 287
 Commentaires • 288
 Le corps de l'application • 288
 Commentaires • 290

17. APPLICATIONS WEB 291

Pages web interactives • 291

Un serveur web en pur Python ! • 292

Première ébauche : mise en ligne d'une page web minimaliste • 294

Ajout d'une deuxième page • 296

Présentation et traitement d'un formulaire • 297

Analyse de la communication et des erreurs • 298

Structuration d'un site à pages multiples • 299

Prise en charge des sessions • 301

Réalisation concrète d'un site web interactif • 303

Le script • 305

Les « patrons » HTML • 314

Autres développements • 317

18. IMPRIMER AVEC PYTHON 319

L'interface graphique peut aider • 320

Le PDF, langage de description de page pour l'impression • 321

Installer Python 2.6 ou 2.7 pour utiliser des modules Python 2 • 322

Exploitation de la bibliothèque ReportLab • 326

Un premier document PDF rudimentaire • 326

Commentaires • 326

Générer un document plus élaboré • 328

Commentaires • 330

Documents de plusieurs pages et gestion des paragraphes • 331

Exemple de script pour la mise en page d'un fichier texte • 332

Commentaires • 334

En conclusion • 336

19. COMMUNICATIONS À TRAVERS UN RÉSEAU**ET MULTITHREADING 339**

Les sockets • 339

Construction d'un serveur rudimentaire • 340

Commentaires • 341

Construction d'un client rudimentaire • 342

Commentaires • 343

Gestion de plusieurs tâches en parallèle à l'aide de threads • 343

Client réseau gérant l'émission et la réception simultanées • 344

Commentaires • 346

Serveur réseau gérant les connexions de plusieurs clients en parallèle • 347

Commentaires • 348

Jeu des bombardes, version réseau • 349

Programme serveur : vue d'ensemble • 350

Protocole de communication • 350

Remarques complémentaires • 352

Programme serveur : première partie • 352

Synchronisation de threads concurrents à l'aide de verrous (thread locks) • 355

Utilisation • 356

Programme serveur : suite et fin • 356

Commentaires • 359

Programme client • 359

Commentaires • 362

Conclusions et perspectives • 363

Utilisation de threads pour optimiser les animations • 363

Temporisation des animations à l'aide de after() • 363

Temporisation des animations à l'aide de time.sleep() • 364

Exemple concret • 365

Commentaires • 366

20. ANNEXE A. INSTALLATION DE PYTHON 369

Sous Windows • 369

Sous Linux • 369

Sous Mac OS • 369

Installation de CherryPy • 370

Installation de pg8000 • 370

Installation de ReportLab et de Python Imaging Library • 371

21. ANNEXE B. SOLUTIONS DES EXERCICES 373**22. INDEX 433**

Gérard Swinnen

De formation scientifique, Gérard Swinnen a enseigné la physique, la chimie et la biologie, et développé une série de logiciels de simulation expérimentale et d'évaluation scolaire. Sollicité pour mettre en œuvre une filière d'enseignement secondaire centrée sur l'apprentissage de l'informatique, il a accepté de construire un cours de programmation spécifiquement adapté à ce public. « Ce que j'affirme, c'est que l'apprentissage de la programmation a sa place dans la formation générale des jeunes, car c'est une extraordinaire école de logique, de rigueur, et même de courage. »

Un livre incontournable pour acquérir l'exigeante discipline qu'est l'art de la programmation !

Original et stimulant, cet ouvrage aborde au travers d'exemples attrayants et concrets tous les fondamentaux de la programmation. L'auteur a choisi Python, langage moderne et élégant, aussi performant pour le développement d'applications web complexes que pour la réalisation de scripts système ou l'analyse de fichiers XML.

Un support de cours réputé et adopté par de nombreux enseignants, avec 60 pages d'exercices corrigés

Reconnu et utilisé par les enseignants de nombreuses écoles et IUT, complété d'exercices accompagnés de leurs corrigés, cet ouvrage original et érudit est une référence sur tous les fondamentaux de la programmation : choix d'une structure de données, paramétrage, modularité, orientation objet et héritage, conception d'interface, multithreading et gestion d'événements, protocoles de communication et gestion réseau, bases de données... jusqu'à la désormais indispensable norme Unicode (le format UTF-8). On verra notamment la réalisation avec Python 3 d'une application web interactive et autonome, intégrant une base de données SQLite. Cette nouvelle édition traite de la possibilité de produire des documents imprimables (PDF) de grande qualité en exploitant les ressources combinées de Python 2 et Python 3.

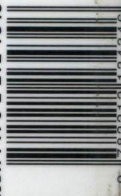
À qui s'adresse ce livre ?

- Aux étudiants en BTS et IUT Informatique et à leurs enseignants ;
- À tous les autodidactes férus de programmation qui veulent découvrir le langage Python.

Au sommaire

Préface. Un support de cours. Choisir un langage de programmation. Distribution de Python. **À l'école des sorciers.** Boîtes noires et pensée magique. Langage machine, langage de programmation. Compilation et interprétation. Mise au point d'un programme. **Données et variables.** Noms de variables et mots réservés. Affectation. Typage. Opérateurs et expressions. Priorité des opérations. Composition. **Contrôle du flux d'exécution.** Séquence d'instructions. Exécution conditionnelle. Opérateurs de comparaison. Blocs d'instructions. Instructions imbriquées. Règles de syntaxe Python. Boucles. Réaffectation. Premiers scripts. **Principaux types de données.** Les listes. **Fonctions.** Interaction avec l'utilisateur. Importer un module de fonctions. Vérité/fausseté d'une expression. Définir une fonction. Variables locales, variables globales. « Vraies » fonctions et procédures. Utilisation des fonctions dans un script. Valeurs par défaut des paramètres. Arguments avec étiquettes. **Interfaces graphiques avec Tkinter.** Programmes pilotés par des événements. Les classes de widgets Tkinter. Contrôler la disposition des widgets. Animation. Récursivité. **Manipuler des fichiers.** Écriture et lecture séquentielle dans un fichier. Gestion des exceptions : les instructions try — except — else. **Approfondir les structures de données.** Chaînes de caractères. Listes. Tuples. Dictionnaires. **Classes, objets, attributs.** Passage d'objets comme arguments. Objets composés d'objets. Objets comme valeurs de retour d'une fonction. **Classes, méthodes, héritage.** La méthode « constructeur ». Espaces de noms des classes et instances. Héritage et polymorphisme. Modules contenant des bibliothèques de classes. **Classes et interfaces graphiques.** Boutons radio. Cadres. Widgets composites. Barres d'outils. Fenêtres avec menus. **Analyse de programmes.** Gestion d'une base de données. SQLite. Ébauche d'un client PostgreSQL. **Applications web.** Pages web interactives. Un serveur web en Python ! Exemple de site web interactif. **Imprimer avec Python.** Utiliser des modules Python 2. Construire des documents PDF avec ReportLab. Documents de plusieurs pages, gestion des images et paragraphes, intégration dans une application web. **Communications à travers un réseau et multithreading.** Les sockets. Construction d'un serveur et d'un client élémentaires. Gérer des tâches en parallèle avec les threads. Connexions de clients en parallèle. Jeu en réseau des bombardes. Des threads pour optimiser les animations. **Installation (Windows, Linux, et Mac OS).** **Solutions des exercices.**

Code article : G13434
ISBN : 978-2-212-13434-6



32 €